INSTITUT FÜR INFORMATIK
DER LUDWIG–MAXIMILIANS–UNIVERSITÄT MÜNCHEN

Diplomarbeit

Anforderungen an Grid-basierte Intrusion Detection Systeme

Christian Maier

Aufgabensteller: Prof. Dr. Heinz-Gerd Hegering

Betreuer: Nils O. v. d. gentschen Felde
Dr. Helmut Reiser

Abgabetermin: 31. März 2008

INSTITUT FÜR INFORMATIK
DER LUDWIG–MAXIMILIANS–UNIVERSITÄT MÜNCHEN

Diplomarbeit

Anforderungen an Grid-basierte Intrusion Detection Systeme

Christian Maier

Aufgabensteller: Prof. Dr. Heinz-Gerd Hegering

Betreuer: Nils O. v. d. gentschen Felde
Dr. Helmut Reiser

Abgabetermin: 31. März 2008

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 31. März 2008

. . . . . .

(Unterschrift des Kandidaten)

Danksagung

Zur Entstehung der vorliegenden Arbeit trugen eine Reihe von Personen bei, denen ich dafür an dieser Stelle meinen Dank aussprechen möchte.

Zuvorderst danke ich meinem Vater *Joachim*, der mich auf meinem Weg bis heute immerzu uneingeschränkt mit Rat und Tat unterstützt hat. Nicht zuletzt durch ihn war es mir überhaupt möglich, mein Studium bis zum abschließenden Punkt der Diplomarbeit voranzubringen.

Des Weiteren möchte ich den Mitgliedern des MNM-Teams um *Prof. Dr. Heinz-Gerd Hegering* danken und hier neben *Dr. Helmut Reiser* vor allem *Nils Otto vor dem gentschen Felde*, der mir von Anbeginn beständig hilfreich zur Seite stand. In vielen Diskussionen und besonders auch durch seine stets konstruktive Kritik konnten die sich wieder und wieder ergebenden Herausforderungen erfolgreich gemeistert werden.

Besonderer Dank geht an *Chris*, die mir trotz großem eigenen beruflichen Engagements abermals wertvolle Anregungen geben konnte. Weiterhin danke ich auch *Rolf Rosenfeldt* für seinen Anteil am guten Gelingen der Arbeit.

Zusammenfassung

Mit der größer werdenden Verbreitung von Grids muss gleichzeitig ein tragfähiges, organisationsübergreifendes Sicherheitsmanagement etabliert werden. Besonders auch im neuartigen und noch stetigem Wandel unterworfenen Umfeld von Grids sind hierfür Intrusion Detection Systeme von großer Bedeutung, wo sie idealerweise etwaige schon bestehende Sicherheitslösungen ergänzen. Mit dem weiteren Anstieg der Leistungsfähigkeit von Grids müssen diese geradezu als „magische“ Anziehungspunkte aufgefasst werden, wobei bedingt durch die charakteristische große Komplexität vielfältige Szenarien missbräuchlicher Nutzung realitätsnah erscheinen. Eine ausgereifte Einbruchserkennung ist nicht zuletzt auch vor dem Hintergrund einer weiter zunehmenden industriellen Nutzung von hohem Wert für eine dauerhafte Überlebensfähigkeit von Grids.

Nach einer grundlegenden Einführung in die Thematik wird ausführlich der Status Quo gridbasierter Ansätze zur Einbruchserkennung untersucht. Es zeigt sich, dass trotz einiger Bemühungen derzeit noch keine umfassende und praktikable Lösung vorgestellt wurde. Vielmehr konzentrieren sich die Ansätze jeweils nur auf einzelne Teilaspekte der gesamten Problematik im Grid, ohne vorher ein angemessenes Fundament und hier besonders die grundsätzlichen Anforderungen im Umfeld von Grids begründet zu haben.

In der Folge widmet sich die Arbeit im Rahmen einer ausführlichen, strukturierten Analyse der Ermittlung wichtiger generischer Anforderungen an gridbasierte Intrusion Detection Systeme, wobei diese jeweils aus praxisnahen Anwendungsfällen abgeleitet werden. Auf diese Weise wird ein Katalog von 48 verschiedenen Anforderungen erarbeitet, die jeweils auch beschrieben und mit einer Erläuterung ihrer Relevanz versehen werden.

Zusätzlich werden mögliche Angriffe auf Grids im Zuge einer grundlegenden Bedrohungsanalyse als Teildisziplin des Security Engineerings diskutiert, wobei vor allem eine klare Klassifizierung derselben auf hohem Abstraktionsgrad erfolgt.

Inhaltsverzeichnis

1 Einführung **1**
- 1.1 Motivation . . . 1
- 1.2 Fragestellung . . . 2
- 1.3 Gliederung der Arbeit . . . 2

2 Grundlagen **5**
- 2.1 Einführung in das Sicherheitsmanagement . . . 6
- 2.2 Intrusion Detection – Zentraler Baustein des Sicherheitsmanagements . . . 7
  - 2.2.1 Fehlerarten . . . 9
  - 2.2.2 Komponenten . . . 9
  - 2.2.3 Einteilung in Typen . . . 10
  - 2.2.4 Techniken zur Erkennung von Angriffen . . . 12
  - 2.2.5 Datenaustauschformate für Intrusion Detection Systeme . . . 15
- 2.3 Grid Computing . . . 18
  - 2.3.1 Geschichte und Entwicklung . . . 18
  - 2.3.2 Begriffsbestimmung . . . 19
  - 2.3.3 Architektur . . . 21
  - 2.3.4 Grid Middlewares – ein Überblick . . . 23
  - 2.3.5 Beispielszenario: D-Grid . . . 27
- 2.4 Besondere Herausforderungen für Intrusion Detection in Grids . . . 28
- 2.5 Zusammenfassung . . . 29

3 Grid-basierte Intrusion Detection: Status Quo **31**
- 3.1 Verteilte Intrusion Detection . . . 32
  - 3.1.1 Anfängliche zentralisierte Ansätze . . . 32
  - 3.1.2 Fortschrittlichere hierarchische Verfahren . . . 35
  - 3.1.3 Aktuelle Projekte . . . 38
- 3.2 Neue, gridspezifische Ansätze . . . 41
  - 3.2.1 Grid-based Intrusion Detection System (GIDS) . . . 41
  - 3.2.2 Grid Intrusion Detection Architecture (GIDA) . . . 42
  - 3.2.3 Performance-based Grid Intrusion Detection System (PGIDS) . . . 43
  - 3.2.4 Integrated Grid-based Intrusion Detection System . . . 44
  - 3.2.5 Weitere Projekte . . . 45
- 3.3 Zusammenfassung . . . 47

4 Analyse zur Gewinnung von wichtigen generischen Anforderungen an gridbasierte Intrusion Detection **49**
- 4.1 Vorgehensweise der Anforderungsanalyse . . . 50
- 4.2 Beschreibung der Aktoren . . . 50
- 4.3 Beschreibung der Anwendungsfälle . . . 52

4.3.1 Organisationsübergreifende Zusammenarbeit . . . 52
4.3.2 Intrusion Detection in Grids . . . 55
4.3.3 Integration in übrige Systeme . . . 57
4.3.4 Ergänzende, grundlegende Anwendungsfälle . . . 57
4.3.5 Übersicht über die Anforderungen . . . 60
4.4 Abgeleitete generische Anforderungen . . . 60
4.5 Zusammenfassung . . . 71

5 Klassifizierung von Angriffen auf Grids 73
5.1 Bedrohungsanalyse: Disziplin des Security Engineering . . . 74
5.2 Frühere Arbeiten . . . 74
5.2.1 Dreigliedriges Schema nach Gerloni et al. [GORP04] . . . 75
5.2.2 Bedrohungsmatrix und -baum nach Eckert [Eck06] . . . 75
5.2.3 Grid-Angriffe nach Schulter et al. [SNKW06] . . . 76
5.2.4 Weitere Ansätze . . . 77
5.2.5 Vergleich der Konzepte . . . 77
5.3 Klassifikation der Angreifer . . . 78
5.4 Klassifikation der Angriffe . . . 80
5.4.1 Bedrohungen durch externe Angriffe . . . 80
5.4.2 Bedrohungen der Datenintegrität und der Informationsvertraulichkeit . . . 81
5.4.3 Bedrohungen der Verfügbarkeit und der Ressourcennutzung . . . 81
5.5 Zusammenfassung . . . 82

6 Zusammenfassung und Ausblick 83
6.1 Zusammenfassung . . . 83
6.2 Offene Fragestellungen . . . 84

Literaturverzeichnis 87

Abbildungsverzeichnis

2.1 Aufbau von IDS 9
2.2 IDMEF-Datenmodell (nach [DCF07], Abschnitt 4.1) 17
2.3 Beispiel eines einfachen Grid-Szenarios 21
2.4 Sanduhrmodell zur Grid-Architektur [Fos02a] 22
2.5 Allgemeine Projektstruktur des D-Grid (nach [D-G]) 28

3.1 Aufbau verteilter, zentralisierter IDS 33
3.2 Aufbau verteilter, hierarchischer IDS 35
3.3 Allgemeine Architektur (nach [BGFI+98], Abbildung 2) 37
3.4 Hierarchische Architektur von DOMINO (nach [YBJ04], Abbildung 1) 39
3.5 Integration des GIDS als virtuelle Organisation (nach [CS03], Abbildung 2) 42
3.6 Aufbau der GIDA (nach [TAWTAS05b], Abbildung 1) 43
3.7 Allgemeiner Aufbau [SNKW06] 45

5.1 Angreifermodell [GORP04] 78

Tabellenverzeichnis

4.1 Übersicht über die Anforderungen und Anwendungsfälle 61

KAPITEL EINS

Einführung

Inhalt des Kapitels

1.1 Motivation 1
1.2 Fragestellung 2
1.3 Gliederung der Arbeit 2

Als Einführung in die vorliegende Diplomarbeit untersucht Abschnitt 1.1 zunächst die Wichtigkeit von Intrusion Detection Systemen gerade in Bezug auf Grids. Darauf folgend wird in Abschnitt 1.2 die konkrete zu bearbeitende Fragestellung sowie in Abschnitt 1.3 die Gliederung der Arbeit vorgestellt.

1.1 Motivation

Mit der größer werdenden Verbreitung von Grids wird gleichzeitig ein tragfähiges, organisationsübergreifendes Sicherheitsmanagement immer wichtiger. Als Ausgangspunkt für die weitere Arbeit soll nun zuerst der grundlegenden Frage nachgegangen werden, warum hierfür die in dieser Arbeit fokussierten Intrusion Detection Systeme im Umfeld von Grids von großer Bedeutung sind.

An dieser Stelle soll bereits auf Abschnitt 2.1 verwiesen werden, in dem dargelegt wird, dass Intrusion Detection Systeme einen maßgeblichen Beitrag für das Management vernetzter Systeme leisten. Dies gilt natürlich genauso und gerade in verteilten Grid-Umgebungen (vgl. Abschnitt 2.3), wo sie sich idealerweise als Komplement zu den übrigen, zumeist stark von der eingesetzten Middleware bestimmten Sicherheitslösungen einfügen [CS03, TAWTAS05b]. Die dringende Notwendigkeit einer systematischen Überwachung auf Angriffe manifestiert sich zudem speziell dadurch, dass wichtige Sicherheitsmechanismen im Grid heute noch Gegenstand reger Forschung sind, und diese daher in vielen Fällen noch nicht voll entwickelt und mit diversen, zum Teil sogar kritischen Problemen behaftet sind [BCR+06].

Darüber hinaus erweisen sich Intrusion Detection Systeme ebenfalls vor dem Hintergrund der für Grids charakteristischen, großen Komplexität als äußerst gewinnbringend. Wie unter anderem aus [Sch00] ersichtlich, gilt die Komplexität schlechthin als größter Feind der IT-Sicherheit. Die sehr heterogenen, interorganisationellen Grid-Topologien implizieren demnach ein gewaltiges Bedrohungspotenzial. Dies gilt besonders auch deshalb, weil sie sich zumeist nur unter Zuhilfenahme von aufwendigen Verfahren absichern lassen, die einerseits kaum frei von Fehlern und andererseits zwangsläufig schwierig zu konfigurieren, einzusetzen und überhaupt zu verstehen sind [BBB04, BWB05].

Einbruchsversuche in Grids müssen wohl mit dem weiteren Anstieg der Leistungsfähigkeit immer mehr als besonders attraktiv und so geradezu als „magische“ Anziehungspunkte aufgefasst werden, was in der Folge gleichfalls eine entsprechend hohe darauf gerichtete Aktivität erwarten lässt

[SNV04]. Die Grids typischerweise inneliegenden hohen Speicher- und Rechenkapazitäten sowie die schnellen Netzanbindungen lassen vielfältige Szenarien missbräuchlicher Nutzung zu, wie beispielsweise den Gebrauch der Infrastruktur zum Brechen kryptografischer Schlüssel, zur Speicherung und Verteilung großer Datenmengen oder zur Durchführung verteilter Denial-of-Service Angriffe. Schon die Kompromittierung einer einzelnen Ressource kann stets den unautorisierten Zugriff auf weitere Daten und Dienste anderer Systeme nach sich ziehen.

In zunehmendem Maße hängen wichtige IT-Prozesse sowohl von Forschungseinrichtungen als auch der Industrie wirtschaftlich direkt oder indirekt von Grids und ergo deren nachhaltiger Verfügbarkeit ab. Daneben ist zu berücksichtigen, dass die zu verarbeitenden Daten nicht selten wertvoll oder vertraulich sind, weshalb damit auch ein erhebliches Schadenspotenzial einhergehen kann. Im Hinblick auf die Verwendung von Grids zur Problemlösung besitzen Sicherheitstechnologien in vielen Anwendungsbereichen generell hinsichtlich der Akzeptanz und Nutzbarkeit einen gewichtigen Stellenwert. Insbesondere sind Intrusion Detection Systeme geeignet, Grids effektiv auf sicherheitskritische Ereignisse zu überwachen und damit das Vertrauen in die Sicherheit der komplexen Infrastruktur zu verbessern. Das Fehlen derartiger Werkzeuge kann hingegen zur vollständigen Ablehnung der Nutzung von Grids führen.

Im Ergebnis bleibt hervorzuheben, dass eine ausgereifte Einbruchserkennung mit einer damit verbunden präventiven Identifikation von Sicherheitslücken von hohem Wert für eine dauerhafte Überlebensfähigkeit von Grids ist, wodurch ihr bei der Realisierung eines beständigen Sicherheitsmanagements konsequent eine tragende Rolle zuteil werden muss.

1.2 Fragestellung

Wie im vorangegangenen Abschnitt aufgezeigt wurde, besteht aktuell und mit zunehmender Bedeutung von Grids besonders auch in Zukunft der dringende Bedarf an gridbasierten Intrusion Detection Systemen. Trotz der offensichtlichen, besonders auch wissenschaftlichen Attraktivität des Problemfeldes, gibt es bis heute nur eine überschaubare Zahl an darauf zielenden Ansätzen, die von einer umfassenden Praxisreife allesamt noch weit entfernt sind. Die Ausarbeitung eines realitätsnahen, ausgereiften Systems für den Einsatz in typischen Grid-Szenarien steht derzeit immer noch aus.

Aufgrund der Mächtigkeit des Themas kann im Rahmen dieser Arbeit nur auf einzelne Teilaspekte der Entwicklung eines gridbasierten Intrusion Detection Systems näher eingegangen werden. Das übrige bleibt gegebenenfalls anderen Autoren für zukünftige Arbeiten überlassen.

Das Ziel dieser Arbeit ist es, im Zuge einer wohl strukturierten Vorgehensweise einerseits die Anforderungen an Einbruchserkennungssysteme im Umfeld von Grids herauszuarbeiten. Dabei muss größter Wert auf ein formalisiertes und strukturiertes Vorgehen gelegt werden, um möglichst sämtliche relevanten Anforderungen ableiten zu können. Zusätzlich soll die Arbeit aber auch schon eine Bedrohungsanalyse abdecken, wobei vor allem eine Klassifikation möglicher Angriffe auf Grids entwickelt werden soll.

1.3 Gliederung der Arbeit

Um die Aufgabenstellung in ihrer ganzen Breite erfassen zu können, wurde für das weitere Vorgehen der vorliegenden Diplomarbeit folgende Systematik angewandt:

In Kapitel 2 wird ein umfassender Überblick zu allen für die weitere Arbeit förderlichen Grundlagen geboten. Nach einer bewusst knapp gehaltenen, aber notwendigen Einführung in das Sicherheitsmanagement beschäftigt sich das Kapitel zunächst mit dem gewichtigen Feld der Intrusion Detection Systeme im Allgemeinen. Im Anschluss daran findet sodann die Problemstellung des Grid Computing breite Berücksichtigung, wobei über die bloße Begriffsbestimmung hinaus vor allem die neuartige Architektur und deren Umsetzung in Form unterschiedlicher Middlewares im Mittelpunkt stehen wird. Ein Beispielszenario rundet das Kapitel schlussendlich ab.

Auf dem Weg zu einer substantiierten Untersuchung der Anforderungen diskutiert Kapitel 3 kritisch den Status Quo von Intrusion Detection Systemen innerhalb des besonderen Grid-Umfeldes. Hierfür ist es unumgänglich, sich zuvorderst mit den vorausgegangenen Konzepten der verteilten Intrusion Detection genauer auseinanderzusetzen, die teilweise richtungsweisendes Wissen hinsichtlich der Übertragbarkeit bestehender Verfahren auf größere Umgebungen brachten. Im weiteren schließt sich dann die Analyse relevanter Forschungsarbeiten zu neuen gridbasierten Intrusion Detection Systemen an, wobei damit insbesondere die dringende Notwendigkeit neuer und progressiver Ideen verdeutlicht werden soll.

Kapitel 4 soll den Kern der Arbeit umfassen, indem nunmehr vorwiegend im Hinblick auf einen planvollen Prozess im Rahmen der Softwareentwicklung von Grund auf die generischen Anforderungen an gridweite Einbruchserkennungssysteme strukturiert ermittelt werden. Aus relevanten, für die Praxis typischen Anwendungsfällen lassen sich wirksam und fundiert die maßgeblichen Anforderungen ableiten. Die auf diese Weise erhobenen, typischen Anforderungen werden in der Folge jeweils separat erklärt und hinsichtlich ihrer Relevanz für die anzustrebende Lösung bewertet.

Das Kapitel 5 führt die Anforderungsanalyse mit der systematischen Aufarbeitung möglicher Angriffe auf Grids fort. Als Teildisziplin des „Security Engineering“ stellt eine solide Bedrohungsanalyse eine fundamentale Basis für jedwedes wirkungsvolle Intrusion Detection System in Grids dar. Nach der Untersuchung möglicher Angreifer wird besonders der Entwicklung einer Klassifikation möglicher Angriffe große Beachtung geschenkt.

Abschließend stellt das letzte Kapitel 6 die im Laufe der Arbeit gewonnenen Informationen noch einmal zusammen und ordnet die gewonnenen Erkenntnisse in den gesamten Kontext der Arbeit. Daneben werden auch offen gebliebene Fragestellungen aufgeführt, deren Aufarbeitung Bestandteil weiterführender Arbeiten sein kann.

KAPITEL ZWEI

Grundlagen

Inhalt des Kapitels

2.1 Einführung in das Sicherheitsmanagement 6
2.2 Intrusion Detection – Zentraler Baustein des Sicherheitsmanagements 7
2.2.1 Fehlerarten 9
2.2.2 Komponenten 9
2.2.3 Einteilung in Typen 10
2.2.3.1 Netzbasierte Intrusion Detection Systeme (NIDS) 10
2.2.3.2 Hostbasierte Intrusion Detection Systeme (HIDS) 11
2.2.3.3 Hybride Intrusion Detection Systeme 11
2.2.4 Techniken zur Erkennung von Angriffen 12
2.2.4.1 Signaturanalyse 12
2.2.4.2 Statistische Analyse, Anomalieanalyse 13
2.2.4.3 Protokollanalyse 14
2.2.4.4 Integritätsanalyse 14
2.2.5 Datenaustauschformate für Intrusion Detection Systeme 15
2.2.5.1 Blocks Extensible Exchange Protocol (BEEP) 15
2.2.5.2 Intrusion Alert Protocol (IAP) 16
2.2.5.3 Intrusion Detection Exchange Protocol (IDXP) 16
2.2.5.4 Intrusion Detection Message Exchange Format (IDMEF) . . . 17
2.2.5.5 Incident Object Description and Exchange Format (IODEF) . . 18
2.3 Grid Computing 18
2.3.1 Geschichte und Entwicklung 18
2.3.2 Begriffsbestimmung 19
2.3.2.1 Was ist ein Grid? 19
2.3.2.2 Virtuelle Organisationen (VOs) 20
2.3.2.3 Nutzungsszenarien 20
2.3.3 Architektur 21
2.3.4 Grid Middlewares – ein Überblick 23
2.3.4.1 Allgemeine Anforderungen 23
2.3.4.2 UNICORE 24
2.3.4.3 Globus Toolkit 25
2.3.4.4 gLite 26
2.3.5 Beispielszenario: D-Grid 27
2.4 Besondere Herausforderungen für Intrusion Detection in Grids 28
2.5 Zusammenfassung 29

Zu Beginn der Diplomarbeit muss die Erarbeitung grundlegenden Hintergrundwissens aus den beteiligten Gebieten stehen. Zur Kerntechnologie der Intrusion Detection Systeme gehört hier insbesondere auch ein Überblick zum derzeitigen Stand der Dinge im Bereich des Grid Computing.

Dieses Kapitel behandelt sodann die folgenden Themen:

Abschnitt 2.1 führt allgemein in das Sicherheitsmanagement ein und stellt wichtige und grundlegende Konzepte vor, um Intrusion Detection Systeme in den übergeordneten Zusammenhang einordnen zu können.

Daran schließt sich in Abschnitt 2.2 ein Überblick Intrusion Detection Systeme betreffend mit Informationen zu den eingesetzten Technologien im Allgemeinen an. Besonderes Augenmerk wird hierbei auf die Einteilung in Typen und die Methoden zur Erkennung von Angriffen gerichtet. Weiterhin werden Formate für den Austausch von sensiblen Informationen zwischen Intrusion Detection Systemen behandelt.

Abschnitt 2.3 befasst sich mit dem relativ neuen Feld des Grid Computing. Hierbei liegt der Schwerpunkt neben der generellen Architektur auch auf Implementierungen in Form von verbreiteten Grid-Middlewares.

Das Kapitel schließt mit Abschnitt 2.4, welcher die besonderen Herausforderungen für Intrusion Detection Systeme in komplexen Grid-Umgebungen untersucht.

2.1 Einführung in das Sicherheitsmanagement

Zum Auftakt des Kapitels soll nachstehend eine kurze Einführung in das Sicherheitsmanagement gegeben werden.

Die Verwaltung von heterogenen Netzwerken setzt die Existenz von Standards und von definierten Schnittstellen voraus. Im Jahr 1989 wurde von der ISO nach beinahe achtjähriger Arbeit im Dokument „ISO/IEC 7498-4“ ein Rahmenmodell, Normen und die Terminologie für das Netzmanagement festgelegt, das sogenannte „OSI Management Framework“. Es gliedert sich in vier Teilbereiche, das Informations-, Organisations-, Kommunikations- und Funktionsmodell. Die zentrale Rolle nimmt das Funktionsmodell ein, das die Gesamtheit der Managementaufgaben wiederum in fünf Funktionsbereiche (engl. *Systems Management Functional Areas*) aufteilt, zu denen neben dem Fehler-, Konfigurations-, Leistungs- und Abrechnungsmanagement auch das im folgenden betrachtete Sicherheitsmanagement gehört.

Das Sicherheitsmanagement soll Netze sowohl gegen Beeinträchtigungen von außen als auch gegen interne Bedrohungen schützen. Dazu bedarf es zunächst einer umfassenden Untersuchung der möglichen Bedrohungen und der damit verbundenen Risiken vernetzter Systeme. Typische Bedrohungen sind beispielsweise [Eck06]:

Passive Angriffe Dazu zählen das Abhören von Datenleitungen (engl. *eavesdropping*) oder das unautorisierte Lesen von Daten. Zurzeit gehören Angriffe, durch die Passworte ausgespäht werden (sog. *Sniffing*), zu den häufigsten passiven Angriffen.

Aktive Angriffe Beispiele für aktive Angriffe sind das Verändern oder Entfernen von Datenpaketen aus einem Nachrichtenstrom, das Wiedereinspielen von Nachrichten oder das unautorisierte Schreiben in Dateien. Dazu gehören auch Maskierungsangriffe (sog. *Spoofing*), bei denen eine falsche Identität vorgegeben wird, und Denial-of-Service Angriffe, die die Verfügbarkeit von Komponenten beeinträchtigen.

Fehlfunktionen von Ressourcen Auch Fehlfunktionen können zu beträchtlichen Gefahren für vernetzte Systeme führen.

Auf Basis der gefundenen Bedrohungen lassen sich eine Reihe von Sicherheitsanforderungen ableiten. Folgende Aufgaben werden miteinbezogen [HAN99]:

- Die Durchführung von **Bedrohungsanalysen** führt zu einer differenzierten Einschätzung des Risikos.
- **Sicherheitsstrategien** (engl. *Security Policies*) dienen der Festlegung und Durchsetzung von Schutzzielen.
- Die **Authentisierung** liefert Gewissheit über Identitäten.
- Das vorrangige Ziel der **Zugriffskontrolle** ist die Verhinderung von nicht autorisierten Operationen auf Rechen- oder Kommunikationssystemen.
- Die Sicherstellung der **Vertraulichkeit** (zum Beispiel mittels Verschlüsselung) soll Informationen vor nicht autorisierten Personen verbergen.
- Durch Sicherung der **Datenintegrität** kann die unautorisierte Veränderung, Löschung, Einfügung, Erzeugung und das Wiedereinspielen erkannt oder verhindert werden.
- Die **Überwachung auf Sicherheitsangriffe** dient der Erkennung sicherheitsrelevanter Ereignisse.
- Schließlich soll die **Berichterstattung zur Sicherheit** (engl. *Reporting*) Meldungen und Warnungen zu Vorfällen ausgeben.

Die im weiteren Verlauf der Arbeit zu behandelnden Intrusion Detection Systeme lassen sich den letzten beiden Punkten „Überwachung auf Sicherheitsangriffe“ und „Berichterstattung zur Sicherheit“ zuordnen.

2.2 Intrusion Detection – Zentraler Baustein des Sicherheitsmanagements

Im vorherigen Abschnitt wurde bereits allgemein in den Bereich des Sicherheitsmanagements eingeführt. Intrusion Detection Systeme sind ein wichtiger Baustein des Sicherheitsmanagements und stellen eine maßgebliche Ergänzung von bereits vorhandenen Sicherheitsstrukturen dar. Sie bieten durch neue Mechanismen zusätzliche Sicherheit und helfen, Angriffe auf die Sicherheitsinfrastruktur zu erkennen und gegebenenfalls zu verhindern. Zurzeit sind Intrusion Detection Systeme noch ein Gebiet, das intensiv erforscht wird.

Vor der Betrachtung der Angriffserkennung (engl. *Intrusion Detection*) muss jedoch erst der Begriff des Angriffs (engl. *Intrusion*) selbst definiert werden, wobei mehrere, unterschiedliche Begriffsbestimmungen aus jeweils unterschiedlichen Blickwinkeln koexistieren. Eine mögliche Definition nach [HML92] beschreibt (Netz-)Angriffe als „jede über das Netz durchgeführte unerwünschte oder unerlaubte Aktivität, die entfernte Ressourcen beeinträchtigt.“ Anderswo [CS95, Eck06] werden Angriffe dagegen als diejenigen Aktivitäten aufgefasst, die die Integrität, Vertraulichkeit oder Verfügbarkeit einer Ressource versuchen zu kompromittieren.

Nachfolgend werden beispielhaft einige Netzangriffe aufgeführt [HML92]:

- Unerlaubte Veränderungen von Systemdateien zum Zwecke des unerlaubten Zugriffs auf System- oder Benutzerinformationen
- Unerlaubter Zugriff auf Benutzerdaten
- Unerlaubte Veränderungen von Benutzerdaten / -informationen
- Unerlaubte Veränderungen von Tabellen oder anderen Systeminformationen in Netzwerkkomponenten
- Unerlaubte Verwendung von Rechnerressourcen (unter anderem durch Erzeugung von neuen Benutzerkonten oder durch die unerlaubte Benutzung von bestehenden Benutzerkonten)

Die Erkennung von Angriffen kann demgegenüber wie folgt definiert werden: „Intrusion Detection ist definiert als Problem, Personen (oder Agenten) zu identifizieren, die ein Computersystem entweder ganz ohne Erlaubnis (zum Beispiel Cracker[1]) verwenden oder aber die legitimen, zugewiesenen Rechte überschreiten (Bedrohung von innen).“ [BGFI+98, MHL94, SBD+91a]

Erste und zugleich fundamentale Ideen auf dem Gebiet der Intrusion Detection stellte Anderson in einem technischen Bericht aus dem Jahr 1980 zur Diskussion [And80]. In den ersten Jahren war die Technologie zur Erkennung von Angriffen noch recht beschränkt. Aber schon wenig später wurde 1987 von Denning ein grundlegend erweiterter Ansatz vorgestellt. Ihre sehr bekannte Veröffentlichung [Den87], die als Grundstein für heutige Intrusion Detection Systeme angesehen wird, beschreibt erstmals modellhaft ein frühes Intrusion Detection System mit dem Ziel, viele verschiedenartige Angriffe zu erkennen. Die Entwicklung von Intrusion Detection Systemen ist demnach im wesentlichen durch vier Faktoren motiviert:

- Die meisten existierenden Systeme besitzen Sicherheitslücken, die sie anfällig für Angriffe oder andere Formen des Missbrauchs machen; sie alle zu finden und zu beheben ist aus technischen und wirtschaftlichen Gründen nicht durchführbar.
- Existierende Systeme mit bekannten Sicherheitslücken werden nicht einfach durch sicherere Systeme ersetzt – hauptsächlich, weil sie interessante Fähigkeiten haben, die in anderen Systemen fehlen, oder aus wirtschaftlichen Gründen.
- Die Entwicklung von Systemen, die „absolut“ sicher sind, ist sehr schwierig, falls nicht sogar unmöglich.
- Sogar die sichersten Systeme sind anfällig für Angriffe von innen, wenn zugewiesene Rechte überschritten werden.

Über die Jahre wurden die damaligen Ideen zu Intrusion Detection Systemen stetig weiterentwickelt und deutlich verbessert. Diverse Prototypen und praxistaugliche Systeme wurden seitdem hervorgebracht, Sobirey [Sob00] listet mittlerweile 92 verschiedene Intrusion Detection Systeme auf. Als ernste Herausforderung hat sich dabei das stark angestiegene Datenaufkommen erwiesen, das von Intrusion Detection Systemen fortwährend produziert wird. Große Anstrengungen wurden seither unternommen, die angefallenen Datenmengen sinnvoll zu korrelieren, zu abstrahieren und zusammenzuführen [MVB03].

[1] Ein Cracker ist ein technisch sehr versierter Angreifer, der jedoch im Unterschied zu einem Hacker einen Angriff gezielt zu seinem eigenen Vorteil oder zum Nachteil eines Dritten durchführt [Eck06].

2.2.1 Fehlerarten

Oftmals ist es für ein Intrusion Detection System problematisch zu unterscheiden, wann nun ein Angriff vorliegt und wann nicht. Dies ist zum einen darin begründet, dass die Menge an vorstellbaren Einfallstoren komplexer Systeme vielfältig und zumindest mit wirtschaftlich vertretbarem Aufwand kaum exakt bestimmbar ist. Zum anderen lassen sich Angriffe oft nicht eindeutig von „normalen“ Zugriffen auf angebotene Dienste trennen. In der Folge muss ein erheblicher Aufwand betrieben werden, um Kriterien für potenzielle Angriffe festzulegen. [GORP04]

Die Güte dieser Kriterien bestimmt direkt den Erfolg und den Nutzen von Intrusion Detection Systemen, was in den letzten Jahren auch eine intensive Forschung in diesem Bereich begründete. Diese Kriterien sind aber alles andere als leicht zu bestimmen, kosten viel Zeit und Geld und können leicht eine schwerwiegende Fehlerquelle darstellen. Sind die Definitionen für Angriffe zu eng gefasst, kann schon die kleinste Änderung im Verhalten des Angreifers ausreichen, damit die Attacke nicht mehr als solche erkannt wird. Werden die Kriterien aber zu weit bemessen, können Verhaltensmuster als Angriffe gedeutet werden, die einem solchen zwar ähneln, aber keiner sind.

Fehlalarme ohne wirklichen Angriff werden falsch positiv (engl. *false positive*) genannt. Nicht erkannte tatsächliche Angriffe nennt man falsch negativ (engl. *false negative*). Jedes Intrusion Detection System erzeugt mehr oder weniger viele falsch positive und falsch negative Meldungen. Die Kunst besteht nun darin, das Intrusion Detection System so zu konfigurieren, dass möglichst wenige solcher Meldungen erzeugt werden. Das ist aufwändig, aber unerlässlich, da bei zu vielen Fehlalarmen wirkliche Angriffe übersehen werden könnten. Falsch negative Meldungen sind bedeutend gefährlicher als falsch positive, da bei diesen ein wirklicher Schaden entstehen kann. Meist wird auf falsch positive Meldungen mehr Aufmerksamkeit verwendet als auf falsch negative, da man die falsch positiven im Unterschied zu den falsch negativen einfacher bemerkt.

2.2.2 Komponenten

Für das weitere Verständnis ist es unerlässlich, sich zu Beginn mit dem Aufbau von Intrusion Detection Systemen zu befassen. Intrusion Detection Systeme besitzen vielfältige Aufgaben, die sich voneinander abgrenzen und kategorisieren lassen. Die Funktionalitäten lassen sich dabei nach [Con02] hauptsächlich in drei Komponenten gliedern (vgl. Abbildung 2.1):

Sensor Beobachtung und Aufzeichnung ausgewählter Parameter des zu überwachenden Systems.
Rudimentäre Vorverarbeitung und nachhaltige Bereitstellung für die weitere Analyse.

Analyse Durchführung einer automatisierten Auswertung.
Insbesondere die Aggregation und Korrelation der von den Sensoren erhaltenen Daten zum Zwecke der Erkennung von Angriffen.

Reporting Aufbereitete Darstellung von während der Analyse erkannten Auffälligkeiten.
Erweiterte Managementaufgaben.
Mechanismen zur Alarmierung.

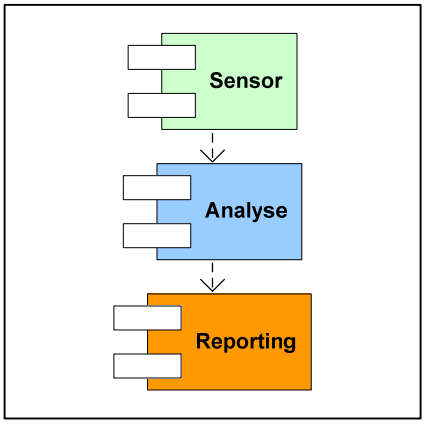

Abbildung 2.1: Aufbau von IDS

Im Zuge dieser Definition leisten Sensoren im Unterschied zu bloßen Agenten neben der reinen Akquise auch noch eine Vorverarbeitung der Daten. Auf diese Weise reduziert sich das Datenaufkommen erheblich, indem Auffälligkeiten nurmehr in Form von „Alarmen“ an die Analysekomponente

weitergereicht werden. Die so gewonnenen Informationen müssen einerseits für die Erkennung von Angriffen ausreichend detailliert sein, andererseits darf ihr Umfang aber auch das nachfolgend analysierende System nicht überlasten.

Nach der Datenkollektion lassen sich Intrusion Detection Systeme in unterschiedliche Typen einteilen, die im anschließenden Abschnitt 2.2.3 beleuchtet werden. Danach behandelt Abschnitt 2.2.4 auch die unterschiedlichen Techniken zur Analyse und Erkennung von Angriffen.

2.2.3 Einteilung in Typen

Intrusion Detection Systeme (IDS) lassen sich nach der Plazierung ihrer Sensoren in verschiedene Typen einteilen. Ein Sensor sammelt Daten und stellt sie zur Auswertung zur Verfügung. Anfangs konzentrierte sich die Entwicklung von Intrusion Detection Systemen noch auf einfachere netzbasierte Verfahren. In der heutigen Zeit decken Intrusion Detection Systeme umfassendere Sensorplazierungen ab. Folgende Typen können unterschieden werden:

Netzbasierte IDS Netzbasierte Intrusion Detection Systeme sind die weitest entwickelte Technologie. Sensoren überwachen ganze Segmente von Netzen. (siehe Abschnitt 2.2.3.1)

Hostbasierte IDS Hostbasierte Intrusion Detection Systeme überwachen jeweils nur einen Host. Zumeist werden Rechner mit erhöhtem Schutzbedarf kontrolliert, wie zentrale Server oder Gateways. Ausgewertet werden Informationen aus System-Logdateien, zur Dateiintegrität oder zur CPU- und Speicherauslastung, um Attacken zu erkennen. (siehe Abschnitt 2.2.3.2)

Hybride IDS Diese Technologie wird auch „Network Node Intrusion Detection“ genannt. Sowohl netzbasierte, als auch hostbasierte Intrusion Detection Systeme werden für einen einzelnen Host integriert. Ein hostbasiertes wird um ein netzbasiertes Intrusion Detection System erweitert, um eine größere Bandbreite an Angriffen abzudecken. (siehe Abschnitt 2.2.3.3)

In den folgenden Abschnitten wird nun auf die verschiedenen Arten von Intrusion Detection Systemen genauer eingegangen. Dabei soll ein besonderes Augenmerk auf den spezifischen Stärken und Schwächen liegen.

2.2.3.1 Netzbasierte Intrusion Detection Systeme (NIDS)

Ein netzbasiertes Intrusion Detection System besteht aus einem oder mehreren Sensoren im Netzwerk und der sie verwaltenden Managementstation. Normalerweise handelt es sich hierbei um eigenständige Rechner, prinzipiell ist aber auch die Zusammenfassung auf nur einem Rechner möglich (zum Beispiel in kleinen Umgebungen). Intrusion Detection Systeme sollten leistungsfähige Rechnersysteme im Netzwerk sein, da zuweilen sehr große Datenmengen in Echtzeit analysiert werden müssen.

Jeder Sensor hört laufend nicht nur die an ihn gerichteten Pakete, sondern den gesamten Netzverkehr ab (engl. *sniffen*) und verarbeitet die Daten dann in einer Analysephase weiter. Aufgefallene Unregelmäßigkeiten werden an die Managementstation gemeldet, die dann für die weitere Aufbereitung und Eskalierung des Alarms sorgt. Die Verlagerung von Analyse- und Entscheidungsprozessen an den Sensor verhindert eine Überlastung der Managementstation.

Gerloni et al. [GORP04] beschreiben unter anderem die folgenden Stärken und Schwächen von netzbasierten Intrusion Detection Systemen:

Da die Sensoren gewöhnlich als zusätzliche Systeme in das Netzwerk integriert werden, ist keine Modifikation an den überwachten Rechnern selbst notwendig und sie werden auch nicht mit zusätzlichen Aufgaben belastet. Im Vergleich zu hostbasierten Intrusion Detection Systemen können durch Einbeziehung mehrerer Sensoren ganze Netze überwacht werden, was insbesondere die Erkennung von verteilten Angriffen erleichtert. Auch wenn eine gute Skalierbarkeit gegeben ist, bleibt die Anzahl möglicher Sensoren letztlich durch die eingeschränkten Betriebsmittel der zentralen Managementstation begrenzt.

Naturgemäß können netzbasierte Intrusion Detection Systeme keine Angriffe innerhalb von verschlüsselten Verbindungen erkennen. Angriffe können weiterhin nur in denjenigen Segmenten des Netzwerks verfolgt werden, denen die Sensoren des Intrusion Detection Systems angehören. Komplexe Analyseverfahren würden sehr hohe Ansprüche an eine Auswertung in Echtzeit stellen, so dass folglich hauptsächlich einfachere Methoden Anwendung finden und so neuartige Angriffe mitunter nicht sicher erkannt werden.

Ein Beispiel für ein netzbasiertes Intrusion Detection System ist das frei erhältliche „Snort“ [Sno].

2.2.3.2 Hostbasierte Intrusion Detection Systeme (HIDS)

Im Falle eines hostbasierten Intrusion Detection Systems ist der Sensor auf dem zu überwachenden Rechner selbst installiert. Dabei werden die auf dem System erzeugten und zur Verfügung stehenden Daten, insbesondere der aktuelle Systemstatus und vorhandene Auditing- und Logging-Mechanismen, analysiert und auf potenzielle Angriffe hin untersucht.

Sie werden typischerweise eingesetzt, um Angriffe zu erkennen, die auf Anwendungs- oder Betriebssystemebene durchgeführt werden. Beispiele für derartige Angriffe sind Überschreitungen der Rechte von Nutzern, Login-Fehlversuche oder Trojaner. [Con02]

Auch die Stärken und Schwächen von hostbasierten Intrusion Detection Systemen werden von Gerloni et al. [GORP04] beleuchtet:

Hostbasierte Intrusion Detection Systeme bilden ein mächtiges Werkzeug für die Analyse von möglichen Angriffen. Aussagekräftige Logdateien erlauben die vollständige Rekonstruktion dessen, was ein Angreifer getan hat. Hostbasierte Intrusion Detection Systeme liefern detailliertere und oft relevantere Daten im Vergleich zu netzbasierten Systemen. Zudem können Angriffe in verschlüsselten Verbindungen erkannt werden, da die Verschlüsselung auf dem Zielsystem endet.

Allerdings ist eine separate Instanz je beobachtetem System zwingende Voraussetzung, was besonders die Beobachtung samt Management ganzer Teilnetze aufwendig und teuer macht. Ereignisse außerhalb der einbezogenen Hosts werden nicht erkannt. Hostbasierte Intrusion Detection Systeme müssen sich auf die vom Betriebssystem bereitgestellten Logging- und Überwachungsmechanismen verlassen, die je nach Betriebssystem differieren. Von einem Eindringling mit unerlaubten Root-Rechten sind sie leicht angreifbar.

Beispiele für Vertreter der Kategorie der hostbasierten Intrusion Detection Systeme sind Tripwire [Tri], Samhain [Sam] oder auch AIDE [AID].

2.2.3.3 Hybride Intrusion Detection Systeme

Hybride Intrusion Detection Systeme vereinen die Vorteile der netz- und der hostbasierten Systeme. Dabei werden die lokal auf dem zu schützenden Rechner verfügbaren Daten mit zusätzlichen Netzinformationen angereichert, um ein umfangreicheres Bild zur Sicherheitslage zu gewinnen. Durch

intelligente Aggregation und Korrelation der auf Ebene des Netzwerks als auch auf der Ebene des Hosts erfassten Daten kann eine deutliche Leistungssteigerung in der Erkennung von Angriffen erreicht werden.

Beispielsweise können Einbruchsversuche auch dann registriert werden, wenn sie im Netzwerk keine oder nur schwer auffindbare Spuren hinterlassen. Genauso können Angriffe erkannt werden, die noch nicht zu Veränderungen auf den Systemen geführt haben und von einem rein hostbasierten Intrusion Detection System daher nicht erkannt würden.

Ein Beispiel für ein bekanntes, frei erhältliches hybrides Intrusion Detection System ist Prelude [Pre].

2.2.4 Techniken zur Erkennung von Angriffen

In den vorausgegangenen Abschnitten wurde bereits eine Klassifizierung und Bewertung von Intrusion Detection Systemen anhand der Positionierung der Sensoren vorgenommen. Daneben gibt es aber auch unterschiedliche Konzepte für die Analyse von Angriffen, die entscheidend zur Erkennungsleistung von Intrusion Detection System beitragen. In den folgenden Abschnitten sollen nur die wichtigsten Techniken mit ihren spezifischen Vor- und Nachteilen beschrieben werden. Eine umfassende Darstellung findet sich unter anderem in [Axe00].

Am weitesten verbreitet sind derzeit Analysemethoden, die auf der Erkennung von Angriffsmustern (der sogenannten „Signaturanalyse“) oder auf der Protokollanalyse beruhen. Die automatische statistische Anomalieerkennung konnte sich noch nicht in auf dem Markt verfügbaren Produkten etablieren. [Con02]

2.2.4.1 Signaturanalyse

Das bekannteste und älteste Verfahren der Angriffserkennung ist die Signaturanalyse. In der Literatur wird die Signaturanalyse auch als „Missbrauchserkennung“ (engl. *Misuse Detection*) geführt.

Zunächst werden aus in der Vergangenheit bereits erfolgten und bekannten Angriffen die sie eindeutig charakterisierenden Merkmale abgeleitet, möglichst genau beschrieben und in ein geeignetes Datenformat gewandelt. Durch Vergleich mit diesen sogenannten „Signaturen“ (oder auch Mustern) trifft dann das Intrusion Detection System die Entscheidung, welches Verhalten erlaubt und welches als Angriff anzusehen ist. Signaturen können sich dabei auf eng umrissene Ereignisse oder auch Prozesse im beobachteten System beziehen.

Der Aufwand für die Erstellung und Pflege von Signaturen[2] ist hoch und folglich auch teuer, zumal Angriffe von großer Komplexität sein können und ständig neue Angriffsszenarien hinzukommen. Für jeden neuen Angriffstypus – auch wenn nur kleine Veränderungen zu einem bereits erfassten Angriff vorliegen – muss immer möglichst zeitnah eine Signatur geschaffen werden, sonst kann der Angriff durch ein signaturbasiertes Intrusion Detection System nicht erkannt werden. Zudem können Signaturen immer erst entwickelt werden, nachdem ein Angriff bekannt wird und vielleicht schon ein beträchtlicher Schaden entstanden ist. Immerhin können zueinander ähnliche Attacken (zum Beispiel die Ausnutzung einer bestimmten Sicherheitslücke) gegebenenfalls in nur einer Signatur zusammengefasst werden. Die Frequenz von Signaturupdates stellt damit insgesamt ein wichtiges Qualitätskriterium dar.

[2] Kommerzielle Intrusion Detection Systeme enthalten einige Hundert bis mehrere Tausend Signaturen (je nach Detailgenauigkeit und Qualität)

Die gesammelten Signaturen werden in einer Datenbank gespeichert und verwaltet, um sie optimiert zur Angriffserkennung heranziehen zu können. Hersteller von Intrusion Detection Systemen liefern zumeist umfangreiche Signaturdatenbanken mit ihren Produkten, die sich in der Regel um weitere Signaturen erweitern lassen. Das Intrusion Detection System analysiert fortwährend Rechner- oder Netzwerkereignisse und untersucht diese auf Entsprechungen im Datenbestand. Aus Übereinstimmungen lässt sich so auf Angriffe schließen.

Die Angriffserkennung nach der Methode der Signaturanalyse gilt heute als weitgehend stabil und ausgereift. Bekannte Angriffe werden sicher und mit einer niedrigen Zahl von falsch positiven Fehlalarmen erkannt. Von Nachteil ist es allerdings, dass prinzipbedingt ausschließlich bekannte Angriffe verfolgt werden können, für die bereits passende Signaturen in der Datenbank vorliegen. Da fortwährend neue Angriffe hinzukommen, führt dies zu einem großen Risiko von falsch negativen Meldungen durch eine potenziell große Anzahl noch nicht berücksichtigter Einbruchstypen.

Eine Weiterentwicklung der einfachen Signaturanalyse stellt die sogenannte zustandsbehaftete Signaturanalyse dar (engl. *Stateful Misuse Detection*). Ein Nachteil der einfachen Signaturanalyse offenbart sich in der Unfähigkeit, fragmentierte Angriffe zu erkennen, bei denen der Datenstrom eines Angriffs unterteilt ist. Durch eine Zustandsmaschine kann dieser Mangel behoben werden, indem zusammengehörige Pakete in der richtigen Reihenfolge gemeinsam analysiert werden. Allerdings bedeutet dieser zusätzliche Aufwand, der sich in einem großem Verbrauch von Ressourcen niederschlägt, im Vergleich zur einfachen Signaturanalyse auch eine gesteigerte Gefährdung durch DoS-Attacken[3].

2.2.4.2 Statistische Analyse, Anomalieanalyse

Mit statistischer Analyse (oder auch Anomalieanalyse) werden Analyseverfahren bezeichnet, bei denen auf Angriffe aus genügend großen Abweichungen von „normalen Verhaltensmustern“ des Systems und seiner Benutzer geschlossen wird. Wie bereits erwähnt, findet sich die statistische Analyse heute noch selten in kommerziellen Intrusion Detection Systemen.

Voraussetzung für die Anwendung dieses Verfahren ist die Erlernung des normalen und damit erlaubten Systemverhaltens. Zu diesem Zweck müssen statistische Daten zu allen relevanten Systemobjekten über einen ausreichend großen Zeitraum gesammelt werden, um daraus ein Profil für die zugelassene Systemnutzung zu erstellen. Solche Parameter können beispielsweise die Anzahl von erfolglosen Logins, die Anzahl von Netzwerkverbindungen oder die Anzahl von Befehlen mit einer Fehlermeldung sein. Durch Vergleich des aktuellen Systemstatus mit dem erlernten Wissen kann dann auf einen Einbruch geschlossen werden. Gegebenenfalls muss das zugrunde liegende Profil auch fortlaufend überprüft und angepasst werden, um berechtigte Änderungen im Verhalten der Anwender aufzunehmen.

Besonders hervorzuheben ist, dass die zur Erlernung des normalen Verhaltens herangezogenen Daten selbst keine Anomalien enthalten sollten. Andernfalls könnte nicht mehr sicher auf Anomalien geschlossen werden. (vgl. [OvdgF05]). Im Gegensatz zur statischen Signaturanalyse, die nur die Erkennung von schon geläufigen Angriffen unterstützt, können mit der statistischen Analyse potenziell auch neue, bislang unbekannte Einbruchstypen aufgedeckt werden. Selbst komplexe Angriffe können entdeckt werden.

Der Ansatz der statistischen Analyse führt leicht zu Problemen in Form von falsch positiven Fehlalarmen, wenn es im Verhalten der Nutzer zu Änderungen kommt, die in der erlernten Statistik noch nicht enthalten sind. Können dem Intrusion Detection System solche Abweichungen nachgetragen

[3] Denial-of-Service: Ausschalten von Diensten

werden, spricht man von adaptiver, andernfalls von statischer Anomalieerkennung. Die adaptive Methode begünstigt allerdings, dass ein Angreifer das Verhalten eines normalen Benutzers über einen längeren Zeitraum so verändert, dass schließlich eine Attacke als normales Benutzerverhalten interpretiert und damit übersehen wird. Systeme, bei denen sich Verhaltensmuster ständig ändern, eignen sich kaum für die statistische Analyse.

Für die Qualität der statistischen Analyse ist es von entscheidender Bedeutung, die Schwellwerte, ab wann eine Verhaltensabweichung als Angriff anzusehen ist, so zu formulieren, dass möglichst wenige falsch positive Fehlalarme ausgelöst werden. Es ist offensichtlich, dass die Bestimmung dieses Grenzwertes schwierig und abhängig von vielen individuellen Faktoren des Systems ist. Immerhin reduziert die statistische Analyse im Vergleich zur Signaturanalyse das Risiko falsch negativer Meldungen.

2.2.4.3 Protokollanalyse

Bei der in heutigen Intrusion Detection Systemen verbreiteten Protokollanalyse wird der gesamte Netzverkehr auf die Einhaltung von genormten Protokollen hin untersucht. Abweichungen von Protokollspezifikationen lassen auf einen versuchten Angriff folgern, zumal gerade im Bereich von IP-basierten Netzen viele Angriffe darauf beruhen, dass von den spezifizierten Protokollen abgewichen wird [Con02].

Im Gegensatz zur signaturbasierten Methode weist die Protokollanalyse eine hohe Performanz auf, da keine Vielzahl an Signaturen abgearbeitet werden muss.

Für den Erfolg des Verfahrens sind exakte Protokollspezifikationen und -implementierungen wichtige Bedingungen, die nicht immer gegeben sind. Schon kleinste Diskrepanzen können zu Fehlalarmen führen. Angriffe, die auf Ungenauigkeiten oder Fehlern in der Protokollspezifikation beruhen, können mitunter nicht erkannt werden.

Um die beschriebenen Defizite auszugleichen und um auch Angriffe zu entdecken, die nicht auf protokollspezifischen Schwachstellen beruhen, ist es unerlässlich, die Erkennungsleistung zusätzlich durch den Einsatz anderer Verfahren (Signaturanalyse etc.) zu verbessern.

2.2.4.4 Integritätsanalyse

Im Vergleich zu den schon aufgeführten Verfahren verfolgt die Integritätsanalyse einen grundlegend anderen Ansatz zur Erkennung von Angriffen. Die relativ einfache Methode findet sich in einer Vielzahl von verfügbaren Intrusion Detection Produkten.

Mithilfe der Integritätsanalyse sollen unzulässige Änderungen (also potenzielle Angriffe) am betrachteten System erkannt werden, indem laufend ausgewählte und für die Systemsicherheit kritische Dateien daraufhin untersucht werden, ob sie seit der letzten Überprüfung unerlaubt verändert wurden. Kann ein Angreifer den Integritätstest selbst deaktivieren, sollte wenigstens das Ausbleiben der Meldungen auf einen Einbruch hindeuten.

Statt etwa jeweils die gesamte Datei zu duplizieren und für die Kontrolle auf Veränderungen heranzuziehen, basiert die Analyse nur auf zuvor in einer Datenbank abgelegten kryptografischen Prüfsummen (engl. *Hashes*), wodurch besonders die Effektivität und Effizienz des Verfahrens stark verbessert wird. Bekannte Algorithmen zur Berechnung von Prüfsummen sind unter anderem „MD5“ [Riv92] und „SHA-1“ [Eas01]. Mit jedem Lauf des Intrusion Detection Systems werden die Prüfsummen der zu überwachenden Dateien neu berechnet und mit den gespeicherten Werten verglichen.

Ergänzend zur Überprüfung auf Abweichungen von der gespeicherten kryptografischen Prüfsumme können auch weitere dateispezifische Metadaten (Datei-, Verzeichniseigenschaften etc.) zur Angriffsanalyse miteinbezogen werden, um die Erkennungsleistung noch zusätzlich zu erhöhen.

2.2.5 Datenaustauschformate für Intrusion Detection Systeme

Betrachtet man Intrusion Detection Systeme, kommt man nicht umhin, sich auch mit den Protokollen und Schnittstellen zwischen den beteiligten Komponenten zu befassen. Für eine reibungslose Kommunikation sind ausgereifte, standardisierte und offene Lösungen unabdingbar.

Dieses Gebiet findet aktuell große Beachtung in der Forschung, so dass immer wieder neue, fortschrittliche Ansätze gefunden werden, um Informationen unter Anwendung geeigneter Methoden und Datenaustauschformate sicher auszutauschen [BSK03]. In letzter Zeit wurden viele solcher Verfahren entwickelt. Dazu gehören das „Intrusion Alert Protocol“ (IAP) [GBF+01], das auf dem „Blocks Extensible Exchange Protocol“ (BEEP) [Ros01] aufbauende „Intrusion Detection Exchange Protocol“ (IDXP) [FM07], das „Intrusion Detection Message Exchange Format“ (IDMEF) [DCF07] sowie das „Incident Object Description and Exchange Format“ (IODEF) [ACDM01, MDD03].

In den folgenden Abschnitten wird jeder dieser Ansätze kurz vorgestellt. Für eine ausführlichere Beschreibung sei ausdrücklich auf die angegebenen Quellen hingewiesen.

2.2.5.1 Blocks Extensible Exchange Protocol (BEEP)

Das „Blocks Extensible Exchange Protocol“ (BEEP) [Ros01] verkörpert ein Framework zur Entwicklung neuer verbindungsorientierter und asynchroner Protokolle der Anwendungsschicht. Es kann auf verschiedene Transportprotokolle abgebildet werden, wobei zumeist TCP [TCP81] zum Einsatz kommt, da es hierfür bereits beispielhaft spezifiziert wurde [Ros01].

Das „Blocks Extensible Exchange Protocol“ stellt dabei selbst noch kein Protokoll für das Versenden oder Empfangen von Daten dar. Es erlaubt vielmehr, eigene Protokolle auf seiner Basis einfacher und zweckmäßiger zu konstruieren, indem verbreitete Mechanismen und Technologien in Form von sogenannten Profilen zur Wiederverwendung bereitgestellt werden. Dazu gehören etwa ein abhörsicherer Datenaustausch, die Authentisierung von Kommunikationspartnern, Integritätsprüfungen von ausgetauschten Nachrichten und viele weitere Bausteine, die in neuen Protokollen nicht abermals implementiert werden müssen.

Von Vorteil ist auch, dass kein festes Datenformat erzwungen wird. Stattdessen kann ein solches passend zu den gegebenen Anforderungen ausgewählt werden (auch binäre Formate). Die Nachrichten werden MIME-kodiert und sind gewöhnlich textuell (gegebenenfalls strukturiert mittels XML).

Der Austausch von Nachrichten erfolgt über sogenannte Kanäle, denen jeweils ein Profil zur Beschreibung von Syntax und Semantik zugeordnet ist. Es werden zwei Arten dieser Kanäle differenziert, nämlich die initialen Kanäle für den Verbindungsaufbau und die kontinuierlichen Kanäle für den Datenaustausch. Über nur eine Verbindung können dabei viele Datenkanäle gleichzeitig abgewickelt werden.

Für das in Abschnitt 2.2.5.3 besprochene „Intrusion Detection Exchange Protocol“ (IDXP) dient das „Blocks Extensible Exchange Protocol“ als Unterbau.

2.2.5.2 Intrusion Alert Protocol (IAP)

Nach Gupta et al. [GBF+01] dient das „Intrusion Alert Protocol“ (IAP) als Protokoll der Anwendungsschicht dem Zecke, aufgelaufene Alarmmeldungen bei Angriffen zwischen den verschiedenen Bausteinen eines Intrusion Detection Systems über IP-basierte Netze auszutauschen. Primär orientiert sich das Protokoll am Austausch von Meldungen zwischen Sensoren, die einen potenziellen Einbruch erkennen, und Managementkomponenten, die den Alarm anzeigen, in eine Datenbank schreiben oder andere vorgesehene Aktionen durchführen. Als Format für die übermittelten Nachrichten wird das „Intrusion Detection Message Exchange Format“ (IDMEF) (vgl. Abschnitt 2.2.5.4) genutzt.

Mit der Entwicklung des Protokolls sollte vor allem das Ziel erreicht werden, notwendige Transport- und Sicherheitsmechanismen bereitzustellen, um die sensiblen Alarmierungen sicher über IP-Netze versenden zu können. Optional kann das Protokoll auch erweitert werden, um zukünftige Entwicklungen, wie die Konfiguration von Sensoren oder das Versenden von Antworten, zu ermöglichen. Als zugrunde liegendes Transportprotokoll kommt das zuverlässige und verbindungsorientierte „Transmission Control Protocol“ (TCP) [TCP81] zum Einsatz.

Der Datenaustausch kommt zumeist direkt zwischen Sensoren und Managementkomponenten zustande, es werden aber auch Verbindungen über Proxies oder Gateways hinweg unterstützt. Verbindungen können dabei sowohl vom Sensor als auch vom Managementbaustein initiiert werden, wobei immer deren zwei – eine je Richtung – benötigt werden. Um den Inhalt von Nachrichten zu kennzeichnen, werden die von E-Mails bekannten „Multipurpose Internet Mail Extensions“ (MIME) (vgl. [BF92, FB96a, FB96b]) verwendet. Die notwendige Verschlüsselung wird schließlich mittels „Transport Layer Security“ (TLS) [DAT+99] sichergestellt.

Ähnlich zum „Intrusion Alert Protocol“ist auch das „Intrusion Detection Exchange Protocol“ (IDXP) (vgl. Abschnitt 2.2.5.3). Das „Intrusion Alert Protocol“ wurde nicht zum de-facto Standard, stattdessen wurde dem Konkurrenten der Vorzug gegeben.

2.2.5.3 Intrusion Detection Exchange Protocol (IDXP)

Wie bereits erwähnt, ähnelt das „Intrusion Detection Exchange Protocol“ (IDXP) [FM07] dem zuvor beschriebenen „Intrusion Alert Protocol“. Es verkörpert ebenfalls ein Protokoll der Anwendungsschicht für den Datenaustausch zwischen Komponenten von Intrusion Detection Systemen.

Das Verfahren unterstützt die gegenseitige Authentisierung, die Sicherung von Integrität und Vertraulichkeit über ein verbindungsorientiertes Protokoll. Als Format für die übermittelten Meldungen ist das für strukturierte Nachrichten geschaffene „Intrusion Detection Message Exchange Format“ (IDMEF) (vgl. Abschnitt 2.2.5.4) vorteilhaft und vorgesehen, wobei der Transport andersartiger Daten, wie unstrukturierter Texte oder binärer Daten, prinzipiell ebenso möglich ist.

Das „Intrusion Detection Exchange Protocol“ basiert auf dem „Blocks Extensible Exchange Protocol“ (BEEP) (vgl. Abschnitt 2.2.5.1). In Teilen ist es dafür als sogenanntes „Profil“ spezifiziert. Dies hat unter anderem den Vorteil, dass mehrere von BEEP bereits in anderen Profilen abgedeckte Funktionen direkt integriert werden können und der Aufwand für eine eigene Implementierung derselben entfällt. Dazu gehören im wesentlichen Mechanismen zur Authentisierung und Sicherung der Vertraulichkeit. Gesichert werden die Daten gleichermaßen unter Verwendung von entsprechenden BEEP-Profilen, beispielsweise durch den „Simple Authentication and Security Layer“ (SASL) oder durch „Transport Layer Security“ (TLS) [DAT+99].

Kommunikationsbeziehungen zwischen den Sensoren und den Managementkomponenten von Intrusion Detection Systemen werden immer paarweise über BEEP etabliert und abgewickelt. Genauso

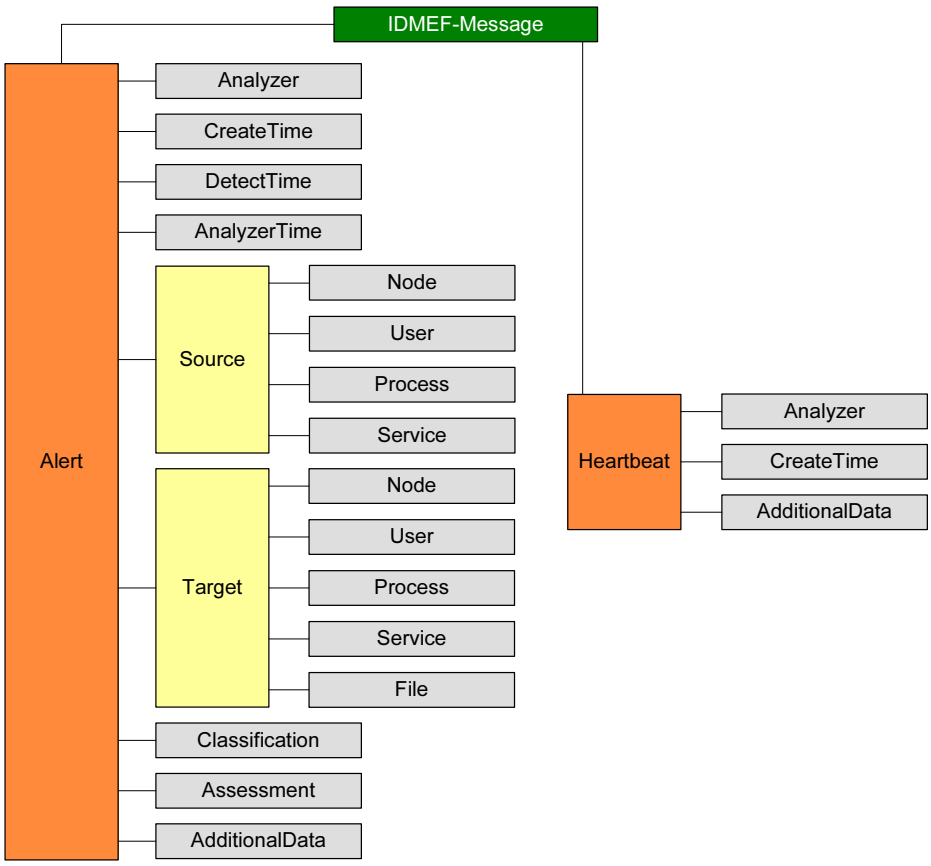

Abbildung 2.2: IDMEF-Datenmodell (nach [DCF07], Abschnitt 4.1)

wie im Falle des „Intrusion Alert Protocol“ können Daten direkt oder über Proxies ausgetauscht werden.

2.2.5.4 Intrusion Detection Message Exchange Format (IDMEF)

Im Gegensatz zu den zuvor beschriebenen Verfahren, die sich neuen Kommunikationsprotokollen für Intrusion Detection Systeme widmen, beschäftigt sich das „Intrusion Detection Message Exchange Format“ (IDMEF) [DCF07] nun hauptsächlich mit dem Inhalt und der Struktur der auszutauschenden Daten. Wie erwähnt, wird es von den beiden Protokollen „Intrusion Alert Protocol“ und „Intrusion Detection Exchange Protocol“ als primäres Datenformat integriert.

Das IDMEF stellt ein auf der Extensible Markup Language (XML) basierendes einheitliches Nachrichtenformat für den Informationsaustausch zwischen Intrusion Detection Systemen dar, wodurch vor allem die Interoperabilität zwischen Produkten unterschiedlicher Hersteller verbessert werden soll. Insbesondere können so heterogene Anwendungen mit ihren individuellen Stärken und Schwächen auch gemeinsam eingesetzt werden, um die Einbruchserkennung zu optimieren.

Das Format ist hauptsächlich für die Kommunikation zwischen Analyse- (oder Sensor-) und Managementkomponenten gedacht, wird aber in der Praxis auch anderweitig eingesetzt (Archivierung in einer Datenbank, Korrelation von Ereignissen, GUI...). Durch den Einsatz von XML lässt es sich leicht weiterverarbeiten und erweitern.

Abbildung 2.2 zeigt das objektorientierte IDMEF-Datenmodell (nach [DCF07], Abschnitt 4.1). Ge-

nerell werden zwei verschiedene Typen von IDMEF-Nachrichten unterschieden, sogenannte Heartbeats und Alerts. Heartbeat-Nachrichten sind sehr einfach aufgebaut und sollen lediglich die Funktionsfähigkeit einzelner Komponenten eines gemeinsamen Intrusion Detection Systems signalisieren. Das Ausbleiben solcher Nachrichten deutet zumeist auf eine Funktionsstörung oder gar einen Ausfall der betreffenden Komponente hin, was wiederum ein Indiz für einen Angriff sein kann. Die komplexeren Alert-Nachrichten dagegen transportieren detaillierte Informationen zu sicherheitsrelevanten Ereignissen, wie etwa zu Quelle, Ziel und Zeitpunkt erkannter Angriffe.

2.2.5.5 Incident Object Description and Exchange Format (IODEF)

Ähnlich dem IDMEF verfolgt das von der „IETF Extended Incident Handling Working Group“ (INCH) vorgebrachte „Incident Object Description and Exchange Format“ (IODEF) [ACDM01] das Ziel, ein standardisiertes Datenformat für die Abbildung von sicherheitsrelevanten Ereignissen aus Intrusion Detection Systemen zu schaffen. Verglichen mit diesem werden die Informationen aber auf einer wesentlich höheren und abstrakteren Ebene beschrieben.

Unternehmen benötigen heute oftmals die Hilfe von externen Stellen, um sich gegen Angriffe zu schützen. Für den hierfür erforderlichen Wissensaustausch wurde bislang auf proprietäre Datenformate zurückgegriffen, die dann vom Empfänger mühsam verarbeitet werden mussten. Während das IDMEF als Format für die Kommunikation zwischen den Komponenten von Intrusion Detection Systemen Anwendung findet, dient das IODEF vorwiegend dem Austausch von Informationen zwischen den für Einbruchsmeldungen verantwortlichen „Computer Security Incident Response Teams“ (CSIRTs) über unterschiedliche administrative Domänen hinweg.

Ein standardisiertes Datenformat reduziert den Verwaltungsaufwand und erleichtert die Zusammenführung von Daten aus unterschiedlichen Quellen. Zudem lassen sich leicht interoperable Anwendungen für das Management erstellen.

Demzufolge liegt der Fokus statt auf der Aufbereitung von umfassenden Rohdaten auch auf einer abstrakten und auf das wesentliche reduzierten Darstellung von Informationen zu Angriffen. Da sich die Anforderungen an Art und Umfang der Auskünfte je nach Anwendungsszenario ändern können, wurde das IODEF als offenes Framework gestaltet, das für spezielle Anwendungsfälle angepasst werden kann. Wie die anderen Technologien stützt sich auch das „Incident Object Description and Exchange Format“ auf XML und erbt so ebenfalls dessen Vorteile. Das Datenmodell beinhaltet neben Informationen zu Angriffen beispielsweise auch Angaben zu Zeitstempeln, Beschreibungen, Bewertungen und Kontaktdaten.

2.3 Grid Computing

2.3.1 Geschichte und Entwicklung

Die ersten Konzepte für die Verteilung von rechen- oder speicherintensiven Anwendungen entstanden bereits in den frühen 1960er-Jahren. Der eigentliche Ursprung modernen Grid Computings wird aber in den Versuchen zu neuartigen Hochgeschwindigkeitsnetzen gesehen, die seit Anfang der 1990er-Jahre hauptsächlich in Forschungseinrichtungen der USA aufkamen [Cat92]. Neben einer Gigabit-Testumgebung an der Universität von Illinois gab es einige weitere Experimente, wie beispielsweise das Forschungsnetz „CASA“, das Supercomputer aus vier Laboratorien in Kalifornien und Neumexiko für aufwendige wissenschaftliche Berechnungen verband.

Zu einer größeren Verbreitung auch bei bislang noch unbeteiligten Wissenschaftlern aus anderen Forschungsbereichen führte schließlich im Jahr 1995 das wegweisende Experiment „Information Wide-Area Year“ (I-WAY) [SWDC97], dessen Ziel ein Verbund von Rechenzentren zur Ausführung von verteilten Anwendungen war. Es wurde auf der Supercomputing Conference 1995 demonstriert und vereinte erstmals 17 Einrichtungen über elf experimentelle Netzwerke quer über die USA und Kanada in einem Hochgeschwindigkeitsnetz, um darüber mehr als 60 Anwendungen zu testen. Wichtiger Bestandteil dieses Projekts war die Entwicklung der rudimentären Software-Plattform „I-Soft“, die grundlegende Managementfunktionen anbot, unter anderem einheitliche Mechanismen für die Zugangskontrolle, die Koordination der Ressourcennutzung und die Durchsetzung von Sicherheitsrichtlinien.

Die neue Idee einer vollständig integrierten Infrastruktur zur gemeinsamen Nutzung heterogener Rechnerressourcen, die einem generellem Paradigmenwechsel gleichkommt, stellte einen großen Unterschied zu den seit langem etablierten Konzepten des verteilten Rechnens dar, wo man sich vornehmlich mit dem Problem der geographischen Verteilung von gleichartigen Ressourcen beschäftigt. I-WAY bildete damit die Basis für weitere bedeutende Arbeiten und heutige Grids.

Insgesamt bleibt festzustellen, dass das gesamte Gebiet des Grid Computings als relativ junge Technologie noch ständigem Wandel unterworfen ist. Das heute bekannte Wissen dazu ist keineswegs als vollständig zu betrachten, vielmehr sind viele Fragestellungen noch wenig untersucht. Das Interesse an einer Weiterentwicklung ist hoch, was sich insbesondere an immer wieder neuen Grid-Projekten zeigt (vgl. [BNP03, GHN+04]).

2.3.2 Begriffsbestimmung

Nachdem bereits auf die Entwicklung von den Ideen des verteilten Rechnens hin zu Grids eingegangen wurde, soll dieser Abschnitt nun versuchen, den Begriff des Grid Computing möglichst genau zu bestimmen, um so auch eine Abgrenzung von verwandten Ansätzen vorzunehmen.

2.3.2.1 Was ist ein Grid?

Waren Grids noch vor einigen Jahren theoretische und beinahe obskure, wissenschaftliche „Gebilde“, so wurden sie zwischenzeitlich doch sehr populär. Wie bereits erwähnt, ist die Entwicklung von Grids noch lange nicht abgeschlossen, was in ständigen Veränderungen sichtbar wird und die Begriffsbestimmung schwierig gestaltet. Was ist nun ein Grid? Um diese Frage zu beantworten, sei zunächst das folgende, bekannte Zitat vorangestellt [FK98]:

> “A computational grid is a hardware and software infrastructure that provides *dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities.*”

Hinter dem Begriff des Grid Computing steckt die zukunftsweisende Idee, die bekannten Konzepte des verteilten Rechnens zu erweitern und zu modernisieren, um so die gestiegenen Anforderungen vorwiegend aus dem Bereich des Höchstleistungsrechnens erfüllen zu können. Ziel ist es, eine neue Infrastruktur bereitzustellen, die Anwendern einen einfachen, transparenten Zugriff auf einen Pool von weltweit über viele Organisationen verteilten Ressourcen ermöglicht. Die Nutzung von Rechen- und Speicherkapazitäten soll dabei genauso einfach möglich sein, wie etwa der Bezug von Elektrizität aus dem Stromnetz [CB02]. Zur Unterscheidung von ähnlichen Ideen sollen die folgenden generellen Kennzeichen von Grids helfen (nach [Fos02b]):

Dezentrale Organisation. Die im Grid verbundenen Ressourcen können sich über viele juristisch, organisatorisch und administrativ autonome und geografisch verteilte Unternehmen erstrecken. Grids unterliegen generell keiner zentralen Kontrolle, was insbesondere das Management erschwert.

Heterogenität. Als Folge der dezentralen Organisation sind Grids oftmals sehr heterogen aufgebaut. Die eingebrachten Ressourcen können sich stark in Hardware, Software und Netzanbindung unterscheiden.

Verwendung standardisierter und offener Protokolle. Nur die Nutzung standardisierter, offener und breit unterstützter Protokolle sowie Schnittstellen als Basis von Grid-Technologien stellt die Interoperabilität innerhalb von komplexen, verteilten Strukturen nachhaltig sicher. Dies dient letztlich auch der Vermeidung von Abhängigkeiten.

Hohe Leistungsfähigkeit. Der koordinierte Zugriff auf integrierte Ressourcen innerhalb von Grid-Verbünden soll zu einem im Vergleich zur Summe der Einzelsysteme signifikant größeren Nutzen und einer erhöhten Qualität des Gesamtsystems beispielsweise hinsichtlich Antwortzeit, Durchsatz, Speicherplatz oder Rechenkapazität führen.

2.3.2.2 Virtuelle Organisationen (VOs)

Ein entscheidender Aspekt bei der Betrachtung des Themas Grid Computing ist die gemeinschaftliche und vor allem auch kontrollierte Nutzung von Ressourcen, dem mit dem innovativen Konzept der sogenannten „Virtuellen Organisationen“ (VOs) begegnet wurde. Zunächst sei wieder ein bekanntes Zitat aufgeführt [FKT01]:

> “The real and specific problem that underlies the Grid concept is *coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations.*”

Virtuelle Organisationen sind dynamische, sich über viele auch geografisch verteilte und administrativ unabhängige Organisationen erstreckende Zusammenschlüsse, um Fähigkeiten und Betriebsmittel, wie Speicherplatz oder Rechenkapazität, zur Lösung eines gemeinsamen Problems zu teilen und so die Leistung insgesamt zu erhöhen.

Ein solches Modell verursacht nicht nur noch nie da gewesene technische Probleme [FK04], sondern birgt in sich auch gewaltige organisationelle Herausforderungen [OvdgFHS06]. Notwendigerweise muss unter den Mitgliedern einer virtuellen Organisation klar definiert werden, auf welche Ressourcen gemeinschaftlich zugegriffen werden darf und welche Bedingungen hierfür Anwendung finden. Es ist anzumerken, dass von den beteiligten Unternehmen dabei nicht nur physische Betriebsmittel (etwa Server) eingebracht werden können, sondern auch logische (etwa Pakete zur Simulation) und sogar personelle Ressourcen (etwa Fachpersonal).

Abbildung 2.3 zeigt beispielhaft ein vereinfachtes Grid-Szenario, in welchem drei Organisationen zwei virtuelle Organisationen bilden. Dabei teilen sie sich heterogene Berechnungs- und Speicherressourcen. Wie in der Abbildung gezeigt, können einzelne Personen oder Betriebsmittel auch an mehreren virtuellen Organisationen beteiligt sein.

2.3.2.3 Nutzungsszenarien

Grids kommen zurzeit mehrheitlich zum Einsatz, um entweder Speicherkapazitäten (sog. *Data Grids*) oder um Rechenkapazitäten (sog. *Computation Grids*) zu erhöhen.

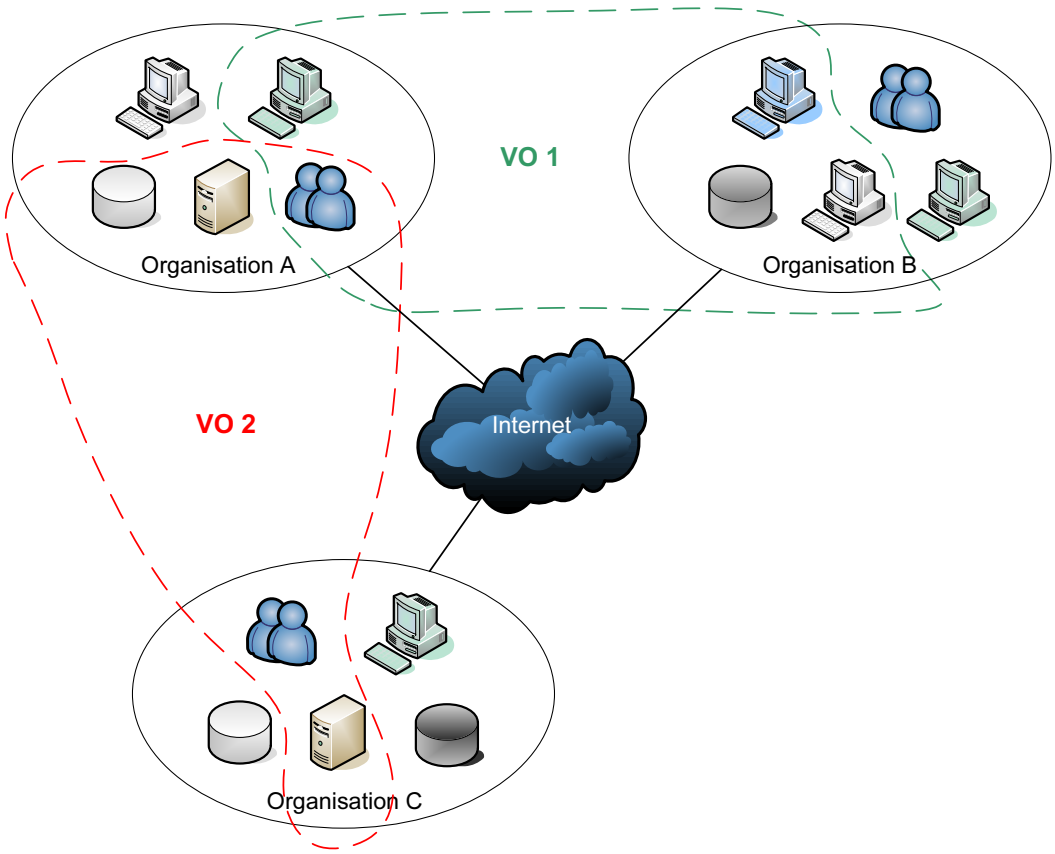

Abbildung 2.3: Beispiel eines einfachen Grid-Szenarios

Damit soll vor allem neuen Herausforderungen des Höchstleistungsrechnens begegnet werden. In vielen Bereichen der Forschung (Hochenergiephysik, Astrophysik, Meteorologie etc.) sieht man sich inzwischen mit Datenmengen konfrontiert, die auf herkömmliche Art und Weise nicht länger zu bewältigen sind. Durch intelligente Verbindung vieler dezentral organisierter Rechner in einem Grid kann dem Problem der Speicherung und dem Austausch von großen Datenmengen begegnet werden. Ebenso lassen sich aufwendige Berechnungen zu anspruchsvollen Aufgaben effizient durchführen, die die Leistungsfähigkeit sonst weit übersteigen würden.

Ein sicherlich bedeutender Grund für die Beschäftigung mit dem Thema Grid Computing ist der Versuch, brachliegende Ressourcen zu nutzen. Eine bessere Auslastung von Hardware unter anderem in Rechenzentren macht teure Investitionen schneller bezahlt.

2.3.3 Architektur

Wie festgestellt, sieht man sich auf dem Gebiet des Grid Computings mit gewaltigen Herausforderungen konfrontiert, wie gerade dem Management von dynamischen, interorganisationellen virtuellen Organisationen. Um diesen zu begegnen, bedarf es neuer Ideen und neuer Technologien. Das Ergebnis eines Jahrzehnts intensiver Forschung an Grids war schließlich ein Konsens über die wesentlichen Anforderungen an Grid-Technologien und deren Architektur.

Im Zuge dieser Entwicklung wurden offene, standardisierte Schnittstellen und Protokolle zur Kommunikation und Steuerung entworfen, die vor allem die wichtige Interoperabilität zwischen verschiedenen Grid-Systemen sicherstellen sollen. Diese Schnittstellen und Protokolle lassen sich ge-

mäß ihrer Funktion in mehrere Schichten aufteilen. Komponenten einer Schicht besitzen ähnliche Eigenschaften und können auf die von niedrigeren Schichten angebotenen Dienste aufbauen. Abbildung 2.4 veranschaulicht das Schichtenkonzept anhand des sogenannten Sanduhrmodells (engl. *Hourglass Model*) [FKT01, Fos02a].

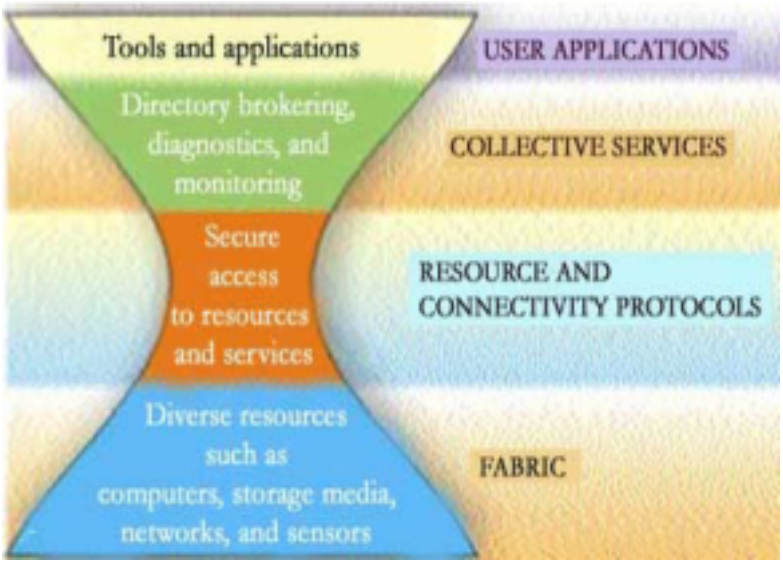

Abbildung 2.4: Sanduhrmodell zur Grid-Architektur [Fos02a]

Ziel des Modells ist es nicht, sämtliche benötigten Protokolle aufzuzählen, sondern Klassen von Komponenten zu identifizieren und deren grundlegende Funktion und Interaktion festzulegen. Das Modell besagt, dass die mittigen Kernschichten nur die nötigsten Funktionen besitzen, während möglichst viel an den Enden der Sanduhr geleistet wird. Dies erleichtert die Portabilität zwischen unterschiedlichen Plattformen, da möglichst wenig neu implementiert werden muss. Der enge Flaschenhals enthält demzufolge nur die Connectivity- und Resource-Schicht, die die grundlegenden Funktionen für den gemeinsamen Zugriff auf Ressourcen bieten. Am unteren Ende liegt die Fabric-Schicht, die einen einheitlichen physischen Zugriff auf die sehr verschiedenartigen Ressourcen im Grid realisiert. Die Collective-Schicht dient schließlich dem Management ganzer Bündel von Ressourcen, auf die von der darüberliegenden Anwendungsschicht zugegriffen werden kann.

Nachfolgend werden die Schichten des Modells detaillierter aufgeführt:

Fabric. Die Fabric-Schicht organisiert den lokalen, jeweils spezifischen Zugriff auf die sehr unterschiedlichen Betriebsmittel, wie Rechner, Speichersysteme, Kataloge, Netze oder Sensoren.

Connectivity. Auf Connectivity-Ebene sind die zentralen Protokolle zur Kommunikation und Authentisierung angesiedelt, die die sichere Interaktion im Grid sicherstellen. Dazu gehören unter anderem Mechanismen für Transport und Routing.

Resource. In die Resource-Schicht fällt die Initiierung, das Monitoring, die Kontrolle und die Abrechnung von einzelnen Zugriffen auf Ressourcen.

Collective. Die Collective-Schicht enthält Protokolle und Dienste, die sich auf gleich mehrere Ressourcen beziehen. Am breiten Ende der Sanduhr gelegen, werden dabei vielfältige Funktionalitäten integriert, wie Verzeichnisdienste, Sicherheits-, Policy- und Abrechnungsmechanismen.

Application. An der Spitze der Sanduhr stehen die eigentlichen Anwendungen, die unter Zuhilfenahme der Protokolle und Dienste der verschiedenen Schichten koordiniert auf die Ressourcen im Grid zugreifen.

2.3.4 Grid Middlewares – ein Überblick

Aufbauend auf den zuvor allgemein erläuterten wesentlichen Konzepten und der Architektur von Grid-Technologien, befassen sich die anschließenden Abschnitte nun mit deren konkreter Umsetzung in Form von unterschiedlichen Grid-Middlewares. Einige dieser Softwarepakete konnten sich behaupten und werden zwischenzeitlich auch schon produktiv eingesetzt.

Gerade in den letzten Jahren wurde die Entwicklung von Grid-Middlewares mit großem Aufwand hauptsächlich von wissenschaftlichen Einrichtungen vorangetrieben. Auch heute noch unterliegt das Gebiet reger Fortentwicklung, was in stetigen Verbesserungen der Systeme resultiert. Nicht zuletzt begünstigt auch das steigende Interesse der Industrie an einer kommerziellen Verwertung das Streben nach ausgereiften Grid-Plattformen.

Da mittlerweile viele Middlewares mit jeweils spezifischen Vor- und Nachteilen entstanden sind, wird im folgenden nach einem Überblick zu allgemeinen Anforderungen beispielhaft nur eine Auswahl verbreiteter Systeme betrachtet. Dazu gehören UNICORE (UNiform Interface to COmputing REsources) [UNI], das Globus Toolkit [Glo] und gLite [gLi]. Diese Middlewares versuchen allesamt eine neue Grid-Infrastruktur einzuführen, die den Zugriff auf verteilte Ressourcen für den Anwender völlig transparent gestaltet.

Im Rahmen dieser Arbeit können die Technologien nur in knapper Form behandelt werden. Für eine umfangreichere Darstellung sei wieder ausdrücklich auf die angegebenen Quellen verwiesen.

2.3.4.1 Allgemeine Anforderungen

Wichtigstes Ziel von Grid-Middlewares ist es, über viele Organisationen verteilte Ressourcen „zusammenzuschalten“ und damit gemeinsam nutzbar zu machen. Die Hauptaufgabe einer jeden Grid-Middleware besteht also darin, den eigentlichen Anwendungen eine einheitliche Schnittstelle für den Zugriff auf die heterogenen Komponenten zur Verfügung zu stellen und dabei die Komplexität von Grids soweit als möglich zu verbergen. Dies dient im wesentlichen einer Vereinfachung in der Entwicklung von Grid-Anwendungen, da die Middleware immer wieder Verwendung finden kann.

Auch wenn es prinzipielle Unterschiede bei Architektur und angewandtem Implementierungsmodell gibt, so überlappen sich die Funktionalitäten der einzelnen Systeme doch in weiten Teilen. Als grundlegende Aufgaben von Grid-Middlewares lassen sich unter anderem identifizieren [LA04]:

- Eine zentrale Funktion von Grid-Middlewares ist die **Ausführung von Programmen** (engl. *Jobs*). Die Middleware entscheidet dabei anhand der momentanen Auslastung und der individuellen Anforderungen, wie (Ort, Reihenfolge etc.) der Auftrag unter Nutzung der zur Verfügung stehenden Ressourcen am besten abgearbeitet werden kann. Abhängigkeiten zwischen den Aufträgen müssen hierbei natürlich berücksichtigt werden. Es muss außerdem immer sichergestellt sein, dass nur berechtigten Nutzern der Zugriff gestattet wird (siehe nächster Punkt).
- Große Bedeutung kommt der **Gewährleistung der Sicherheit** im Grid zu. Stabile Mechanismen zur Authentisierung und Autorisierung von Anwendern sind essenziell und sollten Ressourcen und gegebenenfalls vertrauliche und/oder wertvolle Daten schützen. Eine Verschlüsselung der Kommunikation soll verhindern, dass Daten auf dem Transportweg ausgespäht werden. Gegebenenfalls kann hier auf etablierte Lösungen wie etwa die „Public Key Infrastructure“ (PKI) oder Kerberos [NYHR05] zurückgegriffen werden. Die Größe verteilter Grid-Topologien lässt zudem einen „Single Sign-On“ als wünschenswert erscheinen, damit so die Inanspruchnahme vieler Ressourcen vereinfacht werden kann.

- Von großer Wichtigkeit ist ebenfalls die **Bereitstellung von Informationen** über die beteiligten Ressourcen und Dienste. Dabei kann es sich um Hinweise zu Art und Umfang integrierter Komponenten, zu momentaner Auslastung, verfügbarem Speicherplatz etc. handeln. Die Daten sollten mit Metadaten angereichert in einem sich selbst beschreibenden Format bereitgestellt werden, um auch die Weiterverarbeitung im Sinne einer optimierten Ausführung von Aufträgen durch die Middleware zu unterstützen. Daneben sind die Informationen auch für die Anwender im Grid selbst von Interesse, die darauf über eine vorbereitete Schnittstelle zugreifen können sollten.
- Als weitere Anforderung an Grid-Middlewares ist die **Übertragung von Dateien** anzusehen, die den Austausch auch großer Datenmengen zwischen unterschiedlichen Standorten und hochperformanten Massenspeichersystemen erlaubt. Zusätzlich werden Replikationstechniken benötigt, um Daten aus Gründen der Leistungssteigerung oder zu Backupzwecken an unterschiedlichen Orten vorzuhalten.

Aufbauend auf den elementaren Diensten lassen sich eine Vielzahl von fortschrittlicheren Diensten entwickeln, beispielsweise für das Management von Nutzern oder Aufträgen.

2.3.4.2 UNICORE

Schon vor über zehn Jahren initiierte das heutige Bundesministerium für Bildung und Forschung (BMBF) die Arbeit an „UNICORE“ [UNI], um die Zusammenarbeit und die gemeinsame Nutzung von Ressourcen der Zentren für „High Performance Computing“ (HPC) in Deutschland zu fördern. In den folgenden Jahren wuchs „UNICORE“ zu einer stabilen Middleware-Lösung, die heute an vielen Rechenzentren weltweit im täglichen Betrieb genutzt wird. Die Software ist frei unter der BSD-Lizenz verfügbar und wird regelmäßig weiterentwickelt. Zum aktuellen „UNICORE 6“ trugen unter anderem die Intel Software and Solution Group in Brühl, die Fujitsu Laboratories of Europe in London, die Universität von Warschau, der ICM, CINECA in Bologna und das Forschungszentrum in Jülich bei.

Mit „UNICORE“ soll der Zugriff auf Rechenressourcen vereinheitlicht werden. Bereits der Name – UNICORE ist eine Abkürzung für „UNiform Interface to COmputing REsources“ – weist auf diese Absicht hin. Die Softwareplattform ermöglicht es, nahtlos, sicher und intuitiv auf Ressourcen innerhalb einer verteilten Umgebung zuzugreifen. Verschiedene Hardwarearchitekturen, herstellerspezifische Betriebssysteme, inkompatible Batch-Systeme, unterschiedliche Anwendungsumgebungen oder Sicherheitsrichtlinien werden hierbei vor dem Endanwender verborgen. Die Software zeichnet sich insbesondere durch eine leicht bedienbare, grafische Oberfläche aus. Bei der Entwicklung der Middleware wurde des weiteren auf ausgereifte sicherheitskritische Mechanismen, wie etwa zur Authentisierung von Anwendern, Servern und Software oder zur Verschlüsselung der Kommunikation, und deren Integration in administrative Prozesse geachtet, um einen reibungslosen Betrieb zu gewährleisten. [SEL+05, RM05]

Das UNICORE-Paket basiert auf einem sogenannten „vertikal integrierten“ Softwarestack. Demzufolge sind von der Netzwerkebene bis hin zur Benutzerschnittstelle alle Ebenen der Kommunikation in einem einzigen Programm zusammengefasst. Durch diesen doch sehr monolithischen Ansatz geht Flexibilität verloren, gleichzeitig wachsen aber Umfang und Mächtigkeit des Clients.

Die Vorteile und Funktionen von „UNICORE“ können wie folgt zusammengefasst werden [ABK+05]:

1. **Direkte Erzeugung und Übermittlung von Jobs durch Anwender:** Eine grafische Oberfläche unterstützt den Anwender bei der Generierung komplexer und voneinander abhängiger Jobs, die auf jedem UNICORE-System ohne weitere Änderungen ausgeführt werden können.

2. **Job-Management:** Ein eigenes Job Management System ermöglicht Anwendern die Kontrolle über Jobs und Daten.
3. **Daten-Management:** Während der Erzeugung von neuen Jobs kann der Anwender festlegen, welche Daten zur Ausführung des Jobs importiert oder exportiert werden.
4. **Erweiterbarkeit:** Das User-Interface lässt sich über Plugins erweitern, um speziellen Anforderungen zumeist wissenschaftlicher Anwendungen zu berücksichtigen.
5. **Flusskontrolle:** Ein Job kann als Menge gerichteter und azyklischer Graphen beschrieben werden.
6. **Single Sign-On:** UNICORE bietet einen „Single Sign-On“ mittels X.509-Zertifikaten.
7. **Unterstützung von Legacy-Jobs:** UNICORE unterstützt den traditionellen Batch-Betrieb, indem Anwender auch ihre alten Job-Beschreibungen zu neuen Jobs hinzufügen können.
8. **Ressourcen-Management:** Anwender können Zielsysteme auswählen und benötigte Ressourcen spezifizieren. Der UNICORE-Client überprüft die Korrektheit des Jobs und fordert den Anwender gegebenenfalls zur Korrektur auf.

Einsatz findet „UNICORE“ in einer Vielzahl von Grid-Projekten. Unter anderem wird es vom EUROGRID [EUR] und dessen Anwendungen (BioGrid [Bio], MeteoGrid und CAEGrid), vom „Grid Interoperability Project“ (GRIP) [GRI], vom OpenMolGrid [Opeb] und von der japanischen „National Research Grid Initiative“ (NAREGI) [NAR] als Middleware verwendet.

2.3.4.3 Globus Toolkit

Wie viele andere Grid-Projekte, geht auch das in den späten 1990er-Jahren initiierte „Globus Toolkit“ [Glo] auf das eingangs beschriebene Projekt „I-WAY“ zurück. Anfänglich waren Universitäten und Forschungsinstitute federführend, mittlerweile beteiligen sich aber auch Industriepartner wie IBM, Microsoft oder Sun an der Entwicklung.

Anders als „UNICORE“ stellt die frei verfügbare Grid-Middleware eine reichhaltige Sammlung von grundlegenden Werkzeugen („Bag of Services“) für Grids bereit, die den gemeinsamen Zugriff auf Ressourcen ermöglichen und die auch selektiv genutzt und erweitert werden können. Bestandteil des Pakets sind unter anderem Komponenten betreffend die Sicherheit, das Management von Ressourcen und Daten, die Kommunikation, die Fehlerbehandlung und die Portabilität.

Das „Globus Toolkit“ baut auf der „Grid Security Infrastructure“ (GSI) auf, die ein breites Spektrum an Sicherheitsmechanismen für den Zugriff auf verteilte Ressourcen bereitstellt. Mit den enthaltenen Werkzeugen lässt sich so eine Infrastruktur schaffen, die Anwendern eine sichere Ausführung ihrer Rechenaufträge in einem verteilten und heterogenen Umfeld ermöglicht. Da die „Grid Security Infrastructure“ als Sicherheitsplattform auch in anderen Grid-Middlewares, wie unter anderem bei „gLite“ (siehe Abschnitt 2.3.4.4), sowie unabhängig davon auch in weiteren Anwendungen eingesetzt wird, geht der nächste Abschnitt etwas genauer darauf ein.

Grid Security Infrastructure (GSI) Aufgrund der Komplexität typischer Grid-Szenarien stellt die Entwicklung von Sicherheitslösungen eine große Herausforderung dar. Ressourcen und Anwender sind nicht nur geografisch, sondern auch über Organisationsgrenzen hinweg verteilt, zudem ist ihre Zusammensetzung sehr dynamisch und laufenden Änderungen unterworfen. Eine zusätzliche Anforderung im Grid ist die Delegation von Prozessen, was auch das Starten weiterer Prozesse und das Vererben von Rechten an diese beinhaltet.

Die „Grid Security Infrastructure“ ist eine Sammlung von Sicherheitsprimitiven, Protokollen und APIs[4], die Mechanismen für die besonderen Sicherheitsanforderungen von Grids bieten [Tue01]. Sie lässt sich nach ihren Aufgaben grob in drei logische Teile aufgliedern [FS06]:

1. Sichere, das heißt vertrauliche, authentische und integere Kommunikation wird mit etablierten Verfahren der symmetrischen und asymmetrischen Kryptografie unterstützt, wobei dies sowohl kanal- als auch nachrichtenbasiert geschehen kann.

2. Eine „Public Key Infrastructure“ (PKI) bestehend aus unabhängigen Zertifizierungsstellen (engl. *Certificate Authorities*) garantiert die Identität der Grid-Teilnehmer (Anwender, Systeme und Dienste).

3. Die temporäre Delegation von Identitäts- und Berechtigungsnachweisen (engl. *Credentials*) sowie ein benutzerfreundliches „Single Sign-On“ an Grid-Diensten wird unter Verwendung sogenannter Proxy-Zertifikate und darauf aufbauender Werkzeuge realisiert.

Auf Basis der „Grid Security Infrastructure“ lassen sich weitere Mechanismen zur Unterstützung der Sicherheit im Grid entwickeln, für die noch keine Lösungen enthalten sind.

2.3.4.4 gLite

Als dritte Grid-Middleware soll noch das frei verfügbare „gLite“ [gLi] betrachtet werden, da sie in Europa gerade im wissenschaftlichen Bereich große Beachtung gefunden hat. Sie ging aus dem von der Europäischen Union geförderten Projekt „European DataGrid“ (EDG) [EDG] und besonders aus dem späteren Nachfolgeprojekt „Enabling Grids for E-sciencE“ (EGEE) [EGE] hervor, das unter Einsatz großer finanzieller Mittel erstmals mehr als 70 Institutionen in 27 europäischen Ländern zur Schaffung und Weiterentwicklung einer konsistenten, robusten und sicheren Grid-Infrastruktur verbindet.

Ursprünglich basiert das System auf dem damals schon existenten „LHC Computing Grid“ (LCG) [LCG], das zur Bewältigung der zu erwartenden Datenmengen des aufkommenden „Large Hadron Collider“ (LHC) [LHC] am CERN [CER] in Genf installiert wurde und zu diesem Zweck weit verstreute Ressourcen aus wissenschaftlichen Einrichtungen der Hochenergiephysik zusammenführte. gLite erweitert diesen Ansatz nun hin zu einer neu entwickelten, integrierten Middleware-Lösung, die eine große Bandbreite an grundlegenden gridspezifischen und interoperablen Diensten bietet.

gLite versteht sich dabei als leichtgewichtige Middleware, die eine Vielzahl an anerkannten Bausteinen bestehender Plattformen in sich vereint, um hierdurch eine umfassende und modulare Sammlung an Werkzeugen für alle Bereiche der Nutzung von verteilten Ressourcen im Grid bereitzustellen. Neben Ansätzen aus Projekten wie etwa dem „LHC Computing Grid“ oder „Condor“ [Con] flossen in großen Teilen auch Konzepte und Funktionalitäten aus dem in Abschnitt 2.3.4.3 dargelegten Globus Toolkit (unter anderem die „Grid Security Infrastructure“) ein. gLite kombiniert auf diese Weise eine Kern-Middleware auf niedriger Ebene mit einer Reihe an komponierbaren Diensten auf höherem Niveau.

Ähnlich wie beim Globus Toolkit sind Benutzer nicht gezwungen, das System als Ganzes zu nutzen. Stattdessen können nur jeweils die Teile Anwendung finden, die gerade benötigt werden. Daneben lässt sich das System auch für die individuellen Anforderungen erweitern. Aufbauend auf den zuvor gemachten Erfahrungen wurde bei der Entwicklung vor allem größter Wert auf eine verbesserte Sicherheit und Fortschritte im Management der gesamten Infrastruktur gelegt.

[4] API: Application Programming Interface

Der Einsatz von gLite wird zunehmend auch außerhalb des EGEE-Projekts ernsthaft in Erwägung gezogen, wie etwa im Falle von „DILIGENT“ [Dil], wo sie bereits produktiv läuft, oder auch der französischen Raumfahrtbehörde „CNES“ [CNE].

2.3.5 Beispielszenario: D-Grid

In diesem Abschnitt soll nun das reale Szenario des „D-Grid“ [D-G] beispielhaft einen aktuellen Bezug zur Praxis herstellen, an dessen Aufbau und Betrieb maßgeblich auch das ortsansässige „Leibniz-Rechenzentrum“ (LRZ) [LRZ] beteiligt ist.

Der Grundstein für die D-Grid-Initiative wurde im Januar 2003 gelegt. Bereits im Juli 2003 veröffentlichte eine Gruppe deutscher Wissenschaftler ein richtungsweisendes Strategiepapier [D-G04], in welchem die aktuelle Situation analysiert und der zu erwartende Einfluss des Grid-Computing auf die Forschung in Deutschland untersucht wird. Aufgrund der elementaren Bedeutung wurde als Ergebnis der vorausgegangenen umfangreichen Arbeit ein langfristig ausgerichtetes, strategisches Forschungs- und Entwicklungsprogramm empfohlen.

Diesem Vorschlag folgend begründete das „Bundesministerium für Bildung und Forschung“ (BMBF) [BMB] wenig später, im März 2004, die deutsche D-Grid-Initiative. Zur Förderung von Projekten aus den Bereichen e-Learning, Wissensmanagement und Grid-Computing in Form des anzustrebenden D-Grids sollen in einem Zeitraum von fünf Jahren bis zu 100 Millionen Euro bereitgestellt werden. Unter Zuhilfenahme dieser Infrastruktur wird angestrebt, e-Science-Methoden in der deutschen Wissenschaft zu etablieren, wobei damit ein neuer Ansatz des netzbasierten wissenschaftlichen Arbeitens angedeutet werden soll. Auf der Basis neuester Netztechnologien und in konsequenter Nutzung der Informations- und Wissenstechnologien werden Forschungsprozesse erleichtert, verbessert und intensiviert, indem integrierte Forschungsnetze mit hochleistungsfähigen, verteilten Rechnerressourcen und darauf aufbauenden Diensten innerhalb eines Grids bereitgestellt werden.

Schließlich beteiligten sich im September 2005 neben dem D-Grid-Integrationsprojekt sechs weitere, sogenannte Community-Projekte am Aufbau einer neuartigen, auf Nachhaltigkeit ausgelegten Grid-Infrastruktur. Das Integrationsprojekt nimmt dabei die zentrale Aufgabe wahr, die vielfältigen Entwicklungen der untereinander sehr verschiedenen Community-Projekte in eine gemeinsame D-Grid-Plattform zu integrieren und der deutschen Wissenschaftsgemeinde als Dienste im Kern-D-Grid zugänglich zu machen. Zu den seit Anbeginn beteiligten Communities zählen (vgl. auch Abbildung 2.5):

- **AstroGrid-D:** „German Astronomy Community Grid“ (GACG)
- **C3-Grid:** „Collaborative Climate Community Data and Processing Grid“
- **HEP-Grid:** Grid-Infrastruktur für die Hochenergiephysik
- **InGrid:** Innovative Grid-Entwicklungen für ingenieurwissenschaftliche Anwendungen
- **MediGRID:** Grid-Computing für die Medizin und die Lebenswissenschaften
- **TextGrid:** Community-Grid für die Geisteswissenschaften

Mittlerweile konnten schon 20 Communities in das D-Grid eingebunden werden. In einer zweiten Entwicklungsphase (*D-Grid 2*, Jahre 2007-2010) kommen erweiterte IT-Dienste für Wissenschaft und Industrie hinzu, die auf der Integrationsschicht des D-Grid aufbauen, wie beispielsweise für die Bauindustrie, Finanzwirtschaft, Automobilindustrie, Luft- und Raumfahrt, Betriebsinformations- und Betriebsmittelsysteme und geographische Datenverarbeitung. Für die Zukunft sind noch weitere Schritte zum Ausbau der D-Grid-Infrastruktur geplant.

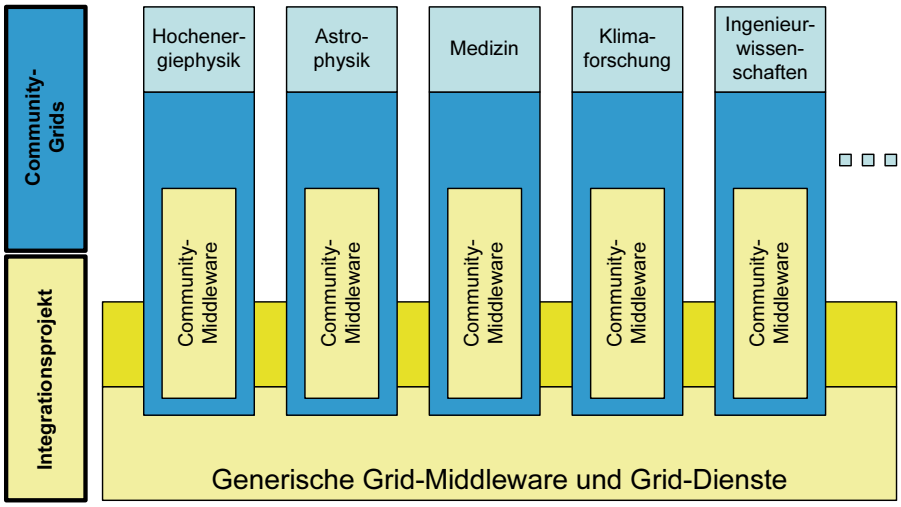

Abbildung 2.5: Allgemeine Projektstruktur des D-Grid (nach [D-G])

2.4 Besondere Herausforderungen für Intrusion Detection in Grids

Wie angesprochen, bedeutete die Weiterentwicklung der Konzepte des verteilten Rechnens hin zu komplexen Grids in besonderem Maße die Erarbeitung frischer und progressiver Herangehensweisen, denen natürlich auch bei der Konzeption von Intrusion Detection Systemen im Umfeld des Grid Computing umfassend Rechnung getragen werden muss. Vielschichtige Fragestellungen, wie die organisationsübergreifende Zusammenarbeit in virtuellen Organisationen und die inhärente Verwendung spezieller Grid-Middlewares für den koordinierten Zugriff auf verteilte Ressourcen, bergen eine Vielzahl an besonderen Herausforderungen, die deshalb fokussiert auf Intrusion Detection Systeme im folgenden genauer analysiert werden sollen.

Eine entscheidende Herausforderung für Intrusion Detection Systeme stellt in jedem Fall das für Grids charakteristische, neuartige Konzept virtueller Organisationen dar (vgl. Abschnitt 2.3.2.2). Neben den inbegriffenen komplexen Vertrauensbeziehungen, die sich hauptsächlich aus der weit gefassten organisatorischen und rechtlichen Autonomie der beteiligten Institutionen ergeben, ist dabei grundsätzlich die hohe Änderungsdynamik ein ernst zu nehmendes Teilproblem. Einerseits ist die dynamische Integration einer möglicherweise sehr großen Anzahl an domänenübergreifenden Ressourcen kennzeichnend, die sich dazu stark in Hardware, Software und Netzanbindung unterscheiden können. Betreffende Organisationen entscheiden stets selbst darüber, wann sie welche und wie viele Ressourcen einbringen wollen. Andererseits unterliegt die ebenso potenziell breite Menge derer, die Zugriff auf die Grid-Ressourcen besitzen, einem fortlaufenden Anpassungsprozess. Nicht übergangen werden sollten gleichfalls diverse Aspekte den Datenschutz betreffend.

Die Zielsetzung eines gridspezifischen Intrusion Detection Systems muss es in erster Linie sein, sich möglichst optimal in die Gegebenheiten des in vielerlei Hinsicht neuen und anspruchsvollen Umfeldes einzupassen und davon im weiteren bestenfalls sogar zu profitieren. Von zentraler Relevanz ist hierbei die jeweils herangezogene Grid-Middleware. Wie in Abschnitt 2.3.4 dargelegt, koexistieren gleich eine ganze Reihe von vergleichbaren Systemen, deren fundamentale Konzepte und ganz besonders auch deren spezifische Sicherheitslösungen trotzdem teilweise erheblich differieren. Beispielshalber seien hier diverse Mechanismen zur Authentifizierung und Autorisierung aufgezählt. Insbesondere wurde bei diesbezüglichen Entwicklungen eine wirkungsvolle Unterstützung von In-

trusion Detection Systemen nicht ausdrücklich bedacht und berücksichtigt, was sich beispielsweise in einer nicht oder kaum vorgesehenen nachhaltigen Aufzeichnung sicherheitsrelevanter Aktivitäten niederschlägt.

Von Interesse sind ebenso die Eigenschaften der Datenübertragung, wie sie sich speziell im Grid-Umfeld zeigen. Generell ist zu konstatieren, dass Grid-Ressourcen zumeist nicht oder nur mit großem Aufwand als solche zu identifizieren sind. Dabei ist einerseits zu erwähnen, dass Ressourcen im Grid häufig über äußerst schnelle Netzanbindungen verfügen und darüber gegebenenfalls riesige Datenmengen übertragen werden, was offensichtlich eine annähernd zeitnahe Auswertung durch Intrusion Detection Systeme erschwert. Daneben hemmt die regelmäßige Verschlüsselung gridspezifischer Daten die durchgängige Kontrolle noch zusätzlich. Auffällig sind außerdem Verbindungsmuster, die deutlich von herkömmlichen Netzverkehr abweichen. So bauen Grid-Ressourcen beispielsweise häufig Verbindungen auf, die über einen sehr langen Zeitraum hinweg fortbestehen.

Alles in allem zeigt sich im Grid eine singuläre Problemstellung, die nolens volens hohe Forderungen an Technologien zur Einbruchserkennung stellt. Es sind daher neuartige Lösungsansätze erforderlich, die nunmehr explizit die spezifischen Eigenschaften von Grids einbeziehen.

2.5 Zusammenfassung

Das vorangegangene Kapitel befasst sich mit den für das Verständnis der weiteren Arbeit elementaren Grundlagen.

Dazu wird zuerst in Abschnitt 2.1 eine kurze, übergreifende Einführung in das Sicherheitsmanagement auf Basis des „OSI Management Frameworks“ gegeben. Dem Sicherheitsmanagement fällt die wichtige Aufgabe zu, Netze sowohl gegen Beeinträchtigungen von außen als auch gegen interne Bedrohungen zu schützen, wozu sich mehrere Sicherheitsanforderungen ableiten lassen.

Abschnitt 2.2 behandelt sodann die Eigenschaften und die verschiedenen Ausprägungen von Intrusion Detection Systemen, wobei netzbasierte, hostbasierte und ebenso hybride Systeme jeweils kurz separat aufgegriffen werden. Außerdem stehen die verschiedenen Techniken zur Erkennung von Angriffen im Vordergrund. Zuletzt ist ein Überblick zu Datenaustauschformaten für Intrusion Detection Systeme Bestandteil des Abschnitts. In letzter Zeit wurden viele solcher Verfahren entwickelt, um Informationen unter Anwendung geeigneter Methoden und Datenaustauschformate gleichfalls domänen- oder organisationsübergreifend sicher auszutauschen.

Abschnitt 2.3 beleuchtet das umfangreiche Thema des Grid Computing, wobei ein besonderes Augenmerk auf der gridtypischen mehrschichtigen Architektur und deren Umsetzung in Form unterschiedlicher Grid-Middlewares liegt. Nach der Untersuchung allgemeiner Anforderungen an Middlewares werden konkret die weit verbreiteten Lösungen UNICORE, das Globus Toolkit und gLite betrachtet. Das naheliegende Beispielszenario des D-Grid soll den Bezug zur Praxis herstellen.

Abschließend stellt Abschnitt 2.4 die besondere Problemstellung bei der Betrachtung von Intrusion Detection Systemen in komplexen Grid-Umgebungen heraus. Die anzustrebende, übergreifende Lösung muss sich bestmöglich in ein anspruchsvolles, hoch dynamisches Umfeld integrieren können.

KAPITEL DREI

Grid-basierte Intrusion Detection: Status Quo

Inhalt des Kapitels

3.1 Verteilte Intrusion Detection 32
3.1.1 Anfängliche zentralisierte Ansätze 32
3.1.1.1 Distributed Intrusion Detection System (DIDS) 32
3.1.1.2 Next-Generation Intrusion Detection Expert System (NIDES) . 34
3.1.2 Fortschrittlichere hierarchische Verfahren 35
3.1.2.1 Graph-based Intrusion Detection System (GrIDS) 36
3.1.2.2 Autonomous Agents For Intrusion Detection (AAFID) 36
3.1.3 Aktuelle Projekte 38
3.1.3.1 Distributed Overlay for Monitoring InterNet Outbreaks (DOMINO) 38
3.1.3.2 Large Scale Intrusion Detection Framework (LarSID) 40
3.2 Neue, gridspezifische Ansätze 41
3.2.1 Grid-based Intrusion Detection System (GIDS) 41
3.2.2 Grid Intrusion Detection Architecture (GIDA) 42
3.2.3 Performance-based Grid Intrusion Detection System (PGIDS) 43
3.2.4 Integrated Grid-based Intrusion Detection System 44
3.2.5 Weitere Projekte 45
3.2.5.1 Grid-specific Host-based Intrusion Detection System (GHIDS) 46
3.2.5.2 Self-adaptive Intrusion Detection System for Computational Grid 46
3.3 Zusammenfassung 47

Dieses Kapitel analysiert den aktuellen Stand der Forschung im relativ jungen Gebiet der gridbasierten Intrusion Detection Systeme.

Abschnitt 3.1 widmet sich hierfür zunächst denjenigen den gridspezifischen Lösungen vorausgegangen Entwicklungen auf dem breiten Gebiet der verteilten Intrusion Detection, die eine maßgebliche Weiterentwicklung vorhandener Ansätze vor allem auch im Hinblick auf den domänenübergreifenden Einsatz darstellt, wie er typischerweise gerade bei verteilten Gridverbünden anzutreffen ist. Dabei sollen kurz die Phasen von den Anfängen bis zum gegenwärtigen Zeitpunkt beschrieben werden.

In Abschnitt 3.2 werden schließlich relevante Forschungsarbeiten zu neuen Grid-basierten Intrusion Detection Systemen analysiert. Es wird besonders darauf eingegangen, warum die bisherigen Arbeiten in weiten Teilen noch wenig ausgereift sind und deshalb neue Konzepte dringend benötigt

werden. Auf dem Weg hin zu einer differenzierteren Lösung beginnt mit dem nächsten Kapitel 4 eine Analyse zur Gewinnung von wichtigen generischen Anforderungen an Intrusion Detection Systeme in Grid-Umgebungen.

3.1 Verteilte Intrusion Detection

Nachdem im vorangegangen Abschnitt 2.4 bereits aufgezeigt wurde, dass die in Abschnitt 2.2 beschriebenen „einfachen“ Intrusion Detection Systeme den Herausforderungen des Einsatzes im Grid keinesfalls genügen können, geht dieser Abschnitt auf die deutlich fortschrittlichere verteilte Intrusion Detection ein. Hierbei partizipieren nun viele Systeme an der Datenkollektion, die nicht unbedingt einer administrativen Domäne angehören und nicht zwingend unter eigener Kontrolle stehen müssen, wie dies gerade auch auf Grids zutrifft. Durch die Zusammenführung von einer großen Menge an verfügbaren Informationen auch über mehrere Netze und sogar Organisationen hinweg verbessern sich das Verständnis von sicherheitsrelevanten Aktivitäten und damit direkt auch die Möglichkeiten bei der Erkennung von großflächig angelegten, verteilten Angriffen.

Aufgrund der großen Breite und Komplexität des Themas gibt es mittlerweile eine Vielzahl an konkurrierenden Lösungsansätzen, von denen im Rahmen dieser Arbeit nur die wichtigsten Varianten in den nächsten Abschnitten grundlegend untersucht werden können. Abschnitt 3.1.1 befasst sich zuerst mit den anfänglichen Ansätzen einer zentral organisierten Aggregation und Korrelation von Informationen verteilt angeordneter Sensoren. Daran knüpft sich in Abschnitt 3.1.2 ein Überblick zur anspruchsvolleren Methode der dezentralen, hierarchisch organisierten Analyse. Schlussendlich werden in Abschnitt 3.1.3 noch einige aktuelle Projekte zusammengefasst.

Für eine detailliertere Darstellung sei wieder ausdrücklich auf die angegebenen Quellen hingewiesen.

3.1.1 Anfängliche zentralisierte Ansätze

Um den massiven Einschränkungen herkömmlicher, einzelner Intrusion Detection Systeme zu begegnen, konzentrierte sich die Forschung anfänglich ausschließlich auf eine Erweiterung der Datenakquise. Die betreffenden Konzepte aus den frühen 1990er-Jahren beinhalten dazu nunmehr eine Menge an hauptsächlich passiven, sich auch über viele Domänen erstreckende Sensoren, die sämtliche beobachteten Informationen ohne zusätzliche Aufarbeitung an einen zentralen Punkt weiterreichen, wo sie anschließend gemeinsam analysiert werden können (vgl. Abbildung 3.1).

Schon bald wurde allerdings klar, dass die auf diese Weise durch viele Sensoren entstehenden großen Datenmengen rasch zu einer Überlastung des analysierenden Systems und des Netzes als Ganzes führen können [SBD+91a, SCCC+96]. Die dadurch stark beeinträchtige Skalierbarkeit kann durch zusätzliche Mechanismen etwa zur Reduktion oder Kompression der Datenflut nur in engen Grenzen verbessert werden, was eine Übertragbarkeit des Ansatzes nur auf kleinere Netze zulässt. Zudem wiegt besonders schwer, dass der Ausfall der zentralen Analysekomponente alleine direkt zu einer vollständigen Unterbrechung der Angriffserkennung führt.

3.1.1.1 Distributed Intrusion Detection System (DIDS)

Mit dem „Distributed Intrusion Detection System“ (DIDS) [BSD+91, SBD+91b] wurde mit Unterstützung durch das „United States Air Force Cryptologic Support Center“ erstmals prototypisch

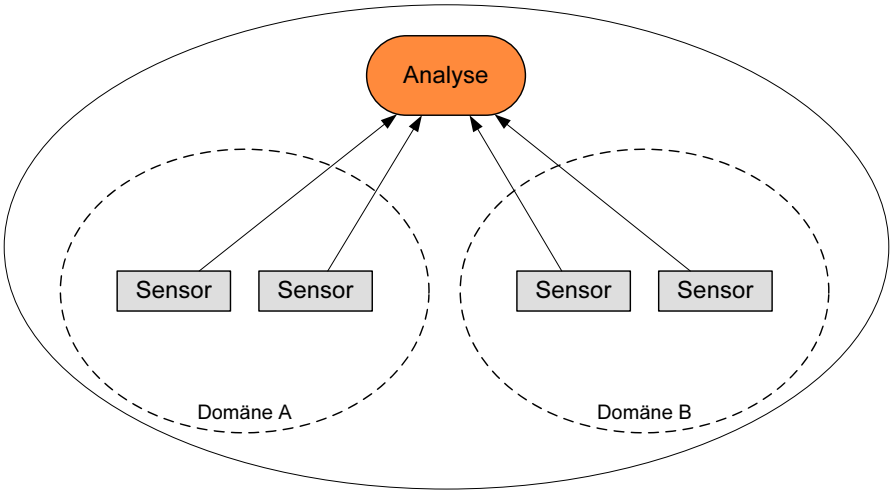

Abbildung 3.1: Aufbau verteilter, zentralisierter IDS

[SSGT92] eine Lösung vorgeschlagen, die verteilte Sensoren mit geeigneter Datenreduktion und eine zentralisierte Auswertung verbindet. Dabei sollen nicht nur einzelne, heterogene Systeme im Netz Berücksichtigung finden, sondern auch das Netz selbst überwacht werden.

Zu den Bausteinen des DIDS zählen folgerichtig die zentrale Analysekomponente, genannt „DIDS-Direktor“. Dazu kommen zum Zwecke der Datenkollektion ein Hostmonitor je einzubeziehendem System und ein Netzmonitor für jedes Segment des Netzes. Der Netzmonitor kann gegebenenfalls auch begrenzt die Aktivität von Systemen ohne eigenen Hostmonitor verfolgen. Die gesammelten Informationen werden von Überwachungskomponenten unabhängig voneinander und asynchron in einem geeigneten Austauschformat weitergegeben, wobei daneben prinzipiell auch eine bidirektionale Kommunikation unterstützt wird, wie sie beispielsweise vom DIDS-Direktor für die Anforderung zusätzlicher Informationen von den Monitoren benötigt würde.

Beide Monitore reduzieren aktiv erheblich das Datenaufkommen, indem sie die anfallenden Informationen rudimentär filtern und nur vorher festgelegte, sicherheitsrelevante Beobachtungen zur Analyse übermitteln. Hostmonitore berücksichtigen hierbei unter anderem fehlgeschlagene Authentisierungen von Benutzern, sicherheitskritische Veränderungen im Systemzustand oder auch Netzzugriffe. Netzmonitore überwachen das Netz auf Abweichungen von vorgegebenen Nutzungsprofilen, wie etwa zu den erlaubten Kommunikationsbeziehungen zwischen den Systemen, zu offen stehenden Diensten oder zum Datenvolumen. Netzmonitore bedienen sich auch erweiterter, heuristischer Methoden, um wahrscheinliche Angriffe erkennen zu können.

Eine der Herausforderungen für Intrusion Detection System besteht darin, Nutzer auch dann eindeutig zu identifizieren, wenn sie sich in Netzen bewegen. Da Angreifer potenziell verschiedene Benutzerkonten auf unterschiedlichen Systemen verwenden, ist es entscheidend, diese wieder einer einzigen Identität zuzuordnen. Das DIDS erzeugt zu diesem Zweck bei der ersten Inanspruchnahme von Ressourcen eine eindeutige, dem Nutzer zugeordnete Kennung (engl. *Network-User Identification (NID)*), die dann gegebenenfalls bei weiteren Aktivitäten in der zu überwachenden Umgebung fortgeschrieben werden kann.

Mit der Entwicklung des DIDS sollte unter anderem zwei wichtigen Bedrohungsszenarien entgegengewirkt werden, nämlich zum einen den sogenannten Türgriffangriffen (engl. *Doorknob Attacks*) und zum anderen dem Durchsuchen von Netzen (engl. *Network Browsing*). In beiden Fällen genügen die zum Zeitpunkt verfügbaren einfachen Intrusion Detection Systeme nicht, um die Angriffe identifizieren zu können.

Das Ziel einer Türgriffattacke liegt darin, ein nicht ausreichend gesichertes System zu entdecken und Zugriff darauf zu erlangen. Dazu probiert der Angreifer gewöhnlich eine begrenzte Anzahl von gängigen Kombinationen aus variierendem Benutzernamen und Passwort auf mehreren Computern, die für eine Erkennung durch die hostbasierten Intrusion Detection Systeme noch nicht ausreichen. Schon solch einfache Angriffe können bemerkenswert erfolgreich sein [Lan85]. Im Falle des Durchsuchens von Netzen durchstöbert ein Nutzer eine große Anzahl an Dateien mehrerer Rechner innerhalb einer kurzen Zeitspanne, deren jeweilige lokale Aktivität wiederum nicht ausreicht, um von einem einzelnen Intrusion Detection System erkannt zu werden.

3.1.1.2 Next-Generation Intrusion Detection Expert System (NIDES)

Nach dem DIDS fand auch das am renommierten „Computer Science Laboratory“ von SRI [SRI] initiierte „Next-Generation Intrusion Detection Expert System“ (NIDES) [AFV95, ALJ+95] als zentralisiert organisiertes verteiltes Einbruchserkennungssystem einige Beachtung. Auf Basis der prototypischen Implementierung von NIDES entstand dort später auch das in vielen Punkten überarbeitete hierarchisch arbeitende System „EMERALD“ [PN97].

Die Architektur von NIDES stützt sich in gleicher Weise auf das Client-Server-Modell. Einerseits wird das aktive Einsammeln der angefallenen Daten von verteilt über mehrere, gegebenenfalls sogar administrativ unabhängige Domänen angeordneten Sensoren und deren zentrale Korrelation durch ein einzelnes selbst nicht überwachtes System, genannt „NIDES-Host“, erbracht. Die Datenakquise übernehmen ausschließlich leichtgewichtige Hostmonitore auf den einzelnen zu überwachenden Komponenten, genannt „Ziel-Hosts“; Netzmonitore waren zumindest angedacht. Da sich die aufgezeichneten Informationen in verschiedenen standardisierten Formaten austauschen lassen, können Hostmonitore prinzipiell plattformübergreifend integriert werden.

Der Kern des Ansatzes besteht darin, mehrere herkömmliche Analysemechanismen in einer verteilten Umgebung gebündelt anzuwenden. Die Korrelation der von den Hostmonitoren aggregierten Daten erfolgt nun parallel gleich auf zweierlei Arten, die sich zur Steigerung der Erkennungsleistung sicherheitsrelevanter Ereignisse ergänzen. Zum einen bedient sich NIDES der fortschrittlichen Anomalieanalyse, um potenziell auch neue, bislang unbekannte Angriffstypen aufzudecken (vgl. Abschnitt 2.2.4.2). Dazu erzeugt es zentral statistische Profile aus dem historischen Nutzungsverhalten der Anwender, die jeweils mit der aktuellen Nutzung abgeglichen werden. Die Auswahl an zu erhebenden Informationen und die Schwellwerte, ab wann eine Abweichung vom historischen Verhalten als Angriff anzusehen ist, sind dabei innerhalb gewisser Grenzen variabel.

Zusätzlich zur Anomalieanalyse obliegt die Untersuchung der akquirierten Daten gleichfalls der Signaturanalyse (vgl. Abschnitt 2.2.4.1) basierend auf der Variante eines regelbasierten Expertensystems. Szenarien zu schon bekannten Angriffen werden hierbei als eine Menge von Regeln kodiert, die eine partiell geordnete Liste der Aktivitäten beobachteter Einbruchsversuche wiedergeben. Das Verfahren ist allerdings insoweit problematisch, dass die Beziehungen zwischen Regeln oft kaum nachzuvollziehen sind und sich die Entwicklung neuer Regeln für komplexe Angriffsszenarien demnach äußerst schwierig gestaltet. NIDES implementiert daher einen performanten modellbasierten Ansatz, bei dem jedes einzubeziehende Szenario separat ausgebildet wird, was in einer übersichtlichen Menge an spezifischen Regeln resultiert. Die Ergebnisse sowohl der Anomalie- als auch der Signaturanalyse werden schließlich zusammengeführt und einheitlich bewertet sowie zur Ausgabe gebracht.

Während das vorherige DIDS die aufgenommenen Daten auf Sensorebene aktiv reduziert, sieht NIDES dagegen keine Vorfilterung vor. Immerhin wird dadurch die Gefahr des Übersehens kritischer Ereignisse eliminiert. Noch mehr als das DIDS leidet das auf eine duale Analyse ausgerichtete NI-

DES aber stark unter der eingeschränkten Skalierbarkeit und zusätzlich auch unter der vollständigen Unterbrechung der Angriffserkennung beim bloßen Ausfall des NIDES-Host.

3.1.2 Fortschrittlichere hierarchische Verfahren

Wie aus den vorangegangenen Abschnitten ersichtlich, leiden die zentralisiert organisierten verteilten Intrusion Detection Systeme an schwerwiegenden Defiziten, die den Einsatz solcher Verfahren im Grunde genommen auf kleinere Umgebungen mit niedrigeren Anforderungen an die Interoperabilität der Komponenten limitieren. Auch die erwähnten Bemühungen, die von den Sensoren beobachteten Daten zur Verbesserung der Skalierbarkeit möglichst effizient auf die für die Angriffserkennung unbedingt benötigten Informationen zu reduzieren, erwiesen sich offensichtlich als schwierig und waren nicht immer zielführend.

Die genannten Umstände führten schließlich zur Weiterentwicklung der Ideen hin zu besser skalierenden, hierarchisch organisierten Verfahren [CS95, BGFI+98]. Zusätzlich zur verteilten Datenkollektion erfolgt hier nun auch noch die Analyse dezentral durch eine größere Anzahl von zwischenliegenden Systemen, die jeweils nur für eine kleinere administrative Domäne des gesamten Netzes zuständig sind. Ihre Informationen zu entdeckten Einbruchsversuchen reichen sie weiter an ein übergeordnetes System, das die Daten der ihm unterstellten Systeme gemeinsam auswertet und dadurch ein umfassenderes Bild des Netzes betreffend die aktuelle Sicherheitslage gewinnt (vgl. Abbildung 3.2).

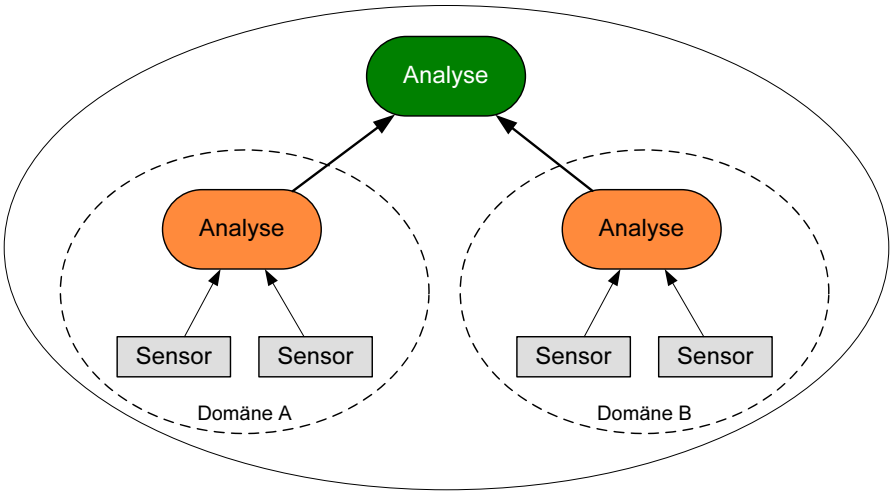

Abbildung 3.2: Aufbau verteilter, hierarchischer IDS

Im Vergleich zu zentral korrelierenden Intrusion Detection Systemen nimmt somit die Zahl der analysierenden Komponenten proportional zur Menge der zu überwachenden Ressourcen zu [SZ00]. Dies bedeutet in der Folge eine weitaus geringere individuelle Belastung und ein deutlich reduziertes Datenaufkommen. Eine der Herausforderungen einer solchen Herangehensweise ist es aber freilich, dass sich großflächig angelegte Angriffe, die von in der Hierarchie weit auseinander liegenden Sensoren aufgezeichnet werden, erst spät auf einem hohen Hierarchieniveau zeigen [AT05].

Die notwendige Partitionierung in die einzelnen Domänen sollte mit Bedacht gewählt werden, da sich hieraus große Optimierungspotenziale ergeben können. Anbieten würde sich eine Einteilung beispielshalber nach der geographischen Lage, nach der administrativen Kontrolle oder auch nach ähnlichen Softwareplattformen der zu überwachenden Systeme [JS99].

3.1.2.1 Graph-based Intrusion Detection System (GrIDS)

Als Einstiegspunkt in die fortschrittlicheren Ansätze hierarchisch arbeitender Intrusion Detection Systeme soll das „Graph-based Intrusion Detection System“ (GrIDS) [SCCC+96, CCD+99] herangezogen werden, für das auch eine prototypische Implementierung existiert. Als eines der ersten Systeme seiner Art war es Vorreiter und Grundlage für neuere Entwicklungen. Es wurde insbesondere geschaffen, um den Flaschenhals der eingeschränkten Erweiterbarkeit zu überwinden und so die Möglichkeiten in der Erkennung von explizit auch großflächig angelegten, verteilten Angriffen weiter zu verbessern.

Die bisher besprochenen Konzepte gründeten allesamt auf einer automatisierten Angriffsanalyse basierend auf den ausschlaggebenden Beobachtungen verteilt angeordneter Sensoren. Im Gegensatz dazu werden durch das GrIDS dynamische Aktivitätsgraphen aggregiert, die lediglich noch Netzkomponenten und die Aktivitäten zwischen ihnen repräsentieren.

Jeder Knoten des Graphen steht dabei für einen oder mehrere zusammengehörige Hosts, die gerichteten Kanten bilden den aktuellen Netzverkehr ab. Knoten und Kanten können mit zusätzlichen Attributen versehen werden. Um ein unbegrenztes Anwachsen von Graphen zu verhindern, werden Knoten bei anhaltender Inaktivität rasch wieder entfernt (was allerdings die Erfassung sehr langsam ablaufender Attacken verhindert). Die Darstellung in Form von Graphen hat den großen Vorteil, dass sie auch vom Menschen leicht aufgenommen werden kann.

Für die Datenakquise verlässt sich das GrIDS auf vorhandene, erprobte Infrastruktur, wie beispielsweise netz- oder hostbasierte Intrusion Detection Systeme. Es besitzt Mechanismen, die eine einfache Erweiterbarkeit um weitere Sicherheitskomponenten sicherstellen, ohne dafür umfangreiche Änderungen erforderlich zu machen.

Um verschiedenartige sicherheitsrelevante Ereignisse unabhängig voneinander analysieren zu können, werden durch das GrIDS gleich eine Anzahl von Graphen etabliert, die jeweils spezifisch für die betrachtete Klasse von Angriffen sind. Zu diesem Zweck lassen sich Graphen auch durch die Einführung von Regelsätzen (engl. *Rule Sets*) flexibel definieren. Sie legen jeweils genau fest, wie die jeweiligen Graphen aus den akquirierten Daten zu konstruieren sind. Zur Angriffserkennung dienen Regeln auch der Beschreibung von typischen Mustern potenziell verdächtiger Graphen, wie sie sich im Falle von Einbruchsversuchen zeigen.

Zur Steigerung der Skalierbarkeit lassen sich Graphen zudem entsprechend der topologischen Gegebenheiten hierarchisch in kleinere Teilbereiche (zum Beispiel administrative Domänen) aufbrechen, die getrennt voneinander verarbeitet werden können. Die lokal ermittelten Informationen fließen dann aggregiert und reduziert zu einem Knoten in die abstraktere Repräsentation des Graphen der darüberliegenden Domäne ein, was dann wiederum die fortgeführte, effiziente Analyse eines größeren Netzsegments zulässt. Die Aufteilung stellt damit vor dem Hintergrund einer drohenden Überlastung beteiligter Systeme sicher, dass keine unakzeptabel dimensionierten Graphen auftreten.

Problematisch ist es, dass konzeptbedingt nur Angriffe verfolgt werden können, die sich auch in der Struktur des Graphen niederschlagen. Neben lokal ausgeführten Attacken bleiben so ebenfalls alle Angriffsversuche unerkannt, deren Netzverkehr unauffällig bleibt.

3.1.2.2 Autonomous Agents For Intrusion Detection (AAFID)

Einen substanziell abweichenden Ansatz verfolgen Balasubramaniyan et al. [BGFI+98], indem sie eine hierarchische Architektur für Intrusion Detection Systeme auf Grundlage voneinander unabhängiger Komponenten, sogenannter autonomer Agenten (vgl. [CS95]), vorschlagen. Das Verfahren

fokussiert dabei auf wesentliche Herausforderungen der Forschung an verteilten Intrusion Detection Systemen, wie die Skalierbarkeit, Performanz und Sicherheit.

Die Entwicklung des neuartigen Konzepts wurde maßgeblich ins Leben gerufen, um auch sehr große Unternehmensnetze aufgeteilt in viele administrative Domänen gezielt und wirkungsvoller als bislang zu schützen. Dazu sieht es einen baumartigen Aufbau mit einer Anordnung der Komponenten in drei logischen, von den physischen Gegebenheiten unabhängigen Schichten vor (vgl. Abbildung 3.3). Die eigentlichen *Agenten*, die die elementare Überwachung auf sicherheitsrelevante Ereignisse erbringen, formen dabei die unterste Ebene. Die Architektur sieht vor, dass mehrere netz- oder hostbasierte Agenten potenziell gleichzeitig einen einzelnen Host beobachten, um so von den individuellen, charakteristischen Vorzügen der jeweiligen Technik zu profitieren. Bemerkenswert ist, dass keine Vorgaben oder Einschränkungen bezüglich der Funktionalität der Agenten gemacht werden.

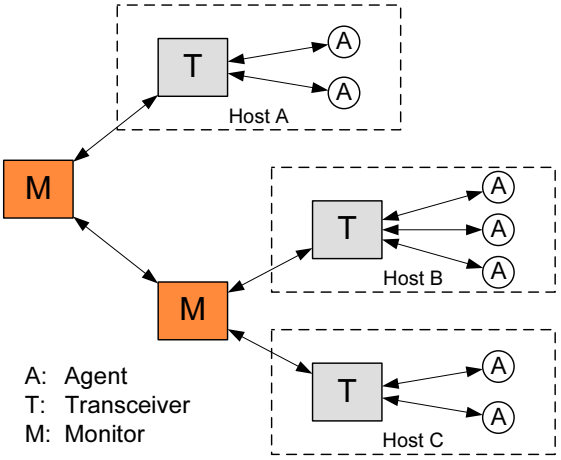

Abbildung 3.3: Allgemeine Architektur (nach [BGFI+98], Abbildung 2)

Agenten kommunizieren nie direkt untereinander. Die von ihnen erzeugten Berichte zu erkannten Auffälligkeiten werden immer an die ihnen übergeordnete Instanz der nächsten Schicht, den sogenannten *Transceiver*, weitergereicht. Ein Transceiver erfasst alle Agenten seines Hosts und nimmt dafür im wesentlichen zwei komplementäre Aufgaben war. Zum einen dient er der Kontrolle, wozu unter anderem Managementanforderungen in Bezug auf die ihm zugehörigen Agenten (beispielsweise das Starten und Beenden, sowie Konfigurationsänderungen) und die nachhaltige Bereitstellung der Daten gehören. Zum anderen dient er der Verarbeitung der von den Agenten erhaltenen Berichte, wobei Informationen auch wieder an Agenten zurückgegeben werden können. Teil dieses Prozesses ist optional auch die gezielte Selektion und Verminderung des Datenumfangs, um Ressourcen einsparen zu können und so die Skalierbarkeit weiter zu stärken.

Auf höchster Ebene residieren die sogenannten *Monitore*. Sie nehmen prinzipiell Aufgaben analog zu den ihnen zugeordneten Transceivern war, diesmal allerdings hostübergreifend. Durch Korrelation der entsprechend bezogenen Informationen lassen sich auch Einbruchsversuche nachvollziehen, die dem Transceiver lokal nicht auffällig erscheinen. Monitore können ihrerseits wiederum andere Monitore bedienen und damit selbst hierarchisch organisiert sein. Für den Fall, dass zwei Monitore denselben Transceiver kontrollieren sollen (etwa aus Gründen der Redundanz), müssen klare Regelungen zur Konsistenzerhaltung existieren.

Die modulare Architektur ermöglicht es, das System auf einfache Weise zu erweitern, zu konfigurieren und zu verändern, entweder durch das Hinzufügen neuer Komponenten oder durch deren Ersetzung. Im Laufe des Projekts zeigte sich, dass effiziente Mechanismen für die Kommunikation von großer Tragweite sind. Speziell der Datenaustausch zwischen den einzelnen Systemen muss zum einen natürlich ausreichend performant sein, gleichzeitig aber auch sehr zuverlässig und sicher. Es wurde hierbei die Frage aufgeworfen, ob ein vorhandenes Protokoll (wie UDP oder TCP) oder stattdessen ein neues, spezifisch für die Anforderungen entworfenes Protokoll Verwendung finden sollte. Angesichts der unterschiedlichen Einsatzszenarien ist dies offensichtlich vom konkreten Anwendungsfall abhängig.

Die Machbarkeit des Verfahrens wurde anhand mehrerer prototypischer Implementierungen ein-

schließlich einer einfachen Benutzerschnittstelle nachgewiesen. Das Konzept leidet aber unter einem zentralen Monitor in der Spitze der Hierarchie, dessen Ausfall nicht trivial zu kompensieren wäre. Es bleibt daher zu klären, wie die aktive Einbruchserkennung effizienter auf die Agenten, Transceiver und Monitore verteilt werden kann, um den Durchsatz und die Skalierbarkeit des Systems zu maximieren, ohne dabei die einbezogenen Hosts und das Netz über Gebühr zu strapazieren.

3.1.3 Aktuelle Projekte

Es ist unbestritten, dass hierarchisch organisierte Verfahren, wie etwa diejenigen des letzten Abschnittes, in vielerlei Hinsicht einen erheblichen Anteil an der Verbesserung der verteilten Intrusion Detection Systeme haben. Durch geeignete Datenreduktion leiten untergeordnete Komponenten jeweils nur noch die für die Angriffserkennung unbedingt erforderlichen Informationen weiter, wo sie dann gemeinsam analysiert werden, um so ein umfassenderes Bild zur Sicherheitslage zu gewinnen. Insbesondere durch eine größere Anzahl zwischenliegender Systeme konnte die Belastung einzelner Systeme und das Datenaufkommen maßgeblich reduziert werden.

Allerdings ist all dieses Ansätzen gemein, dass ihr Erfolg letztlich auf einer Anzahl einzelner, dedizierter Knoten beruht, die jeweils für die vollständige Analyse des ihnen unterstellten Teils des gesamten zu überwachenden Szenarios aufkommen müssen. Obgleich hierarchische Strukturen und die Vorfilterung bereits auf niedrigem Niveau die Möglichkeiten der Systeme vor allem in Bezug auf die Erkennung großflächig angelegter Angriffe potenziell deutlich steigern konnten, limitieren solche Knoten doch nach wie vor, freilich auf höherer Stufe, die Skalierbarkeit. Die Übertragbarkeit der Konzepte speziell auf sehr große Umgebungen erscheint daher problematisch. Zudem sind mehrschichtige Strukturen offensichtlich sehr verwundbar, was Fehlfunktionen oder die Überlastung von Knoten anbetrifft, die in der Hierarchie weit oben nahe der Wurzel liegen. Ihr Ausfall führt direkt zur Eliminierung eines großen Teils der Angriffserkennung (nämlich des gesamten Teilbaums).

Im Rahmen dieses Abschnitts sollen jetzt einige neuere Verfahren näher beleuchtet werden, die versuchen, die gezeigten Probleme im Sinne einer noch besser skalierenden und gegenüber Ausfällen einzelner Komponenten resistenteren Lösung zu überwinden.

3.1.3.1 Distributed Overlay for Monitoring InterNet Outbreaks (DOMINO)

Ein mögliches Konzept für die Lösung der vielschichtigen Probleme von über viele administrative Domänen verteilter Intrusion Detection wurde im Rahmen des vielbeachteten Projekts „Distributed Overlay for Monitoring InterNet Outbreaks“ (DOMINO) [YBJ04] vorgestellt. Im Vergleich zu den dedizierten Elementen vorheriger Modelle ermöglicht die veränderte Organisation der Komponenten als nunmehr kooperierende Partner eines Overlay-Netzes[1] einen äußerst skalierbaren Datenaustausch, die Integration heterogener Systeme sowie ein hohes Maß an Fehlertoleranz und Robustheit.

DOMINO bedient sich dazu auf Basis von verteilten Hashtabellen (engl. *Distributed Hash Tables (DHTs)*) der Vorzüge eines Peer-to-Peer (P2P) Ansatzes. Ein solches Schema ermöglicht vollständig dezentralisierte, skalierbare und sogar selbst-organisierende Systeme, die sich selbsttätig an Veränderungen der Infrastruktur anpassen können. Hierbei ist es von entscheidendem Vorteil, dass alle Komponenten für eine ähnliche Anzahl an Daten verantwortlich sind und somit bereits von vornherein gleichmäßig belastet werden. Wäre die Beanspruchung unausgeglichen, würde dies zu einer möglichen Verminderung der Bandbreite und einem Ansteigen der Reaktionszeit einiger Knoten führen, wohingegen andere unterbeschäftigt bleiben.

[1] Ein Overlay-Netz setzt logisch auf ein schon bestehendes Netz auf [SS02].

Wie viele andere aktuelle Entwicklungen verlässt sich auch DOMINO auf Chord [SMK+01] als eine mögliche, ausgereifte Instantiierung des P2P-Modells. Chord kann hierbei als ein erweiterter Routingdienst angesehen werden, der einem vorhandenen Netz übergeordnet wird. Konsistentes Hashing [KLL+97] sichert dabei die dauerhafte Stabilität des Systems und stellt sicher, dass jederzeit eindeutige Schlüssel auf alle Zielknoten des Overlay-Netzes zeigen.

Der hierarchische Aufbau von DOMINO sieht drei Schichten vor (vgl. Abbildung 3.4). Zentraler Bestandteil der Architektur sind dabei diejenigen Knoten der höchsten Ebene, genannt Achsenknoten (engl. *Axis Nodes*). Jeder dieser Achsenknoten hält eine lokale und dazu eine globale Sicht aller sicherheitsrelevanten Aktivitäten aufrecht. Die lokale Sicht bezieht dabei Ereignisse im eigenen Netz und seinen Satelliten ein. In regelmäßigen Abständen werden zudem Daten von den übrigen Achsenknoten empfangen, die sodann in die Erstellung der globalen Sicht einfließen. Um verteilte Angriffe erkennen zu können, müssen die verschiedenen Achsenknoten ihre Aktivitäten untereinander koordinieren und kooperieren.

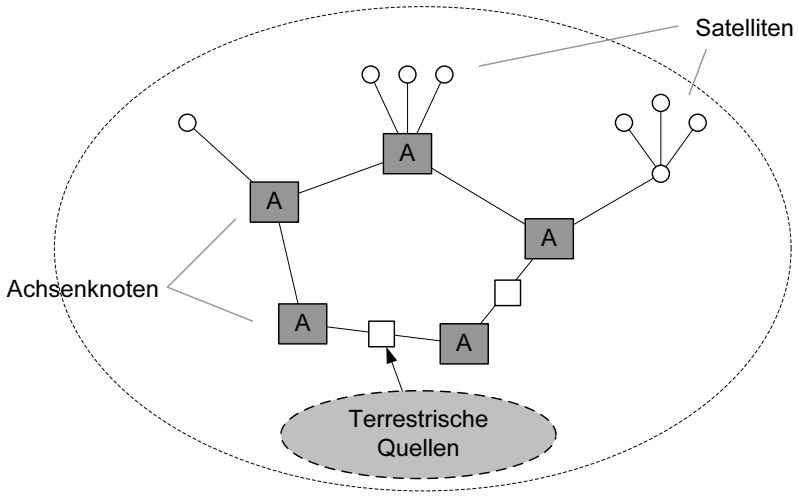

Abbildung 3.4: Hierarchische Architektur von DOMINO (nach [YBJ04], Abbildung 1)

Da Achsenknoten für den Großteil des Austausches sicherheitsrelevanter Informationen verantwortlich zeichnen, ist ihre Skalierbarkeit und dauerhafte Verfügbarkeit von entscheidender Bedeutung für den Gesamterfolg des Systems. Insbesondere können durch den Einsatz eines ausfallsicheren Overlay-Netzes diesbezügliche Anforderungen erfüllt werden [ABKM01]. Zum einen stellt es sicher, dass ein fehlerhafter Achsenknoten nicht gleich einen völligen Verlust der Einbruchserkennung nach sich zieht. Zum anderen erlaubt es die rasche Anpassung an topologische Veränderungen infolge hinzukommender oder wegfallender Knoten.

Auf nächster Ebene finden sich die sogenannten Satelliten (engl. *Satellite Communities*), die ihrerseits ein kleineres Netz unterhalb der Achsen- oder anderer Satellitenknoten aufspannen. Satelliten sind selbst ebenfalls hierarchisch organisiert, so dass Knoten ihre Informationen entweder an einen übergeordneten Satelliten- oder aber direkt an einen Achsenknoten weiterleiten. Die gesamte Kommunikation unter Achsenknoten sowie zwischen Achsen- und Satellitenknoten erfolgt auf Basis einer Kombination von Push- und Pullmechanismen immer authentisiert und verschlüsselt. Von Satelliten bezogene Informationen werden als weniger vertrauenswürdig angesehen.

Eine potenziell große Menge an für eine Einbruchserkennung relevanten Daten liefern zudem weitere, noch weniger vertrauenswürdige terrestrische Quellen (engl. *Terrestrial Contributers*). Da der Ansatz keine bestimmten Anforderungen an sie stellt, kommen hierfür eine Vielzahl an Systemen in Betracht. Durch sie lässt sich die Datenbasis weiter verbreitern und somit auch die Effizienz des

Intrusion Detection Systems verbessern.

Für einen normalisierten Datenaustausch kommt ein auf dem „Intrusion Detection Message Exchange Format“ (vgl. Abschnitt 2.2.5.4) basierendes Protokoll zum Einsatz. Dies stellt die Interoperabilität unterschiedlichster Knoten sicher und erlaubt daneben die maximale Erweiterbarkeit hinsichtlich zukünftiger Weiterentwicklungen.

3.1.3.2 Large Scale Intrusion Detection Framework (LarSID)

Nach dem zuvor betrachteten DOMINO soll dieser Abschnitt mit dem „Large Scale Intrusion Detection Framework“ (LarSID) [ZKL07, ZKL05] auf ein weiteres, ähnlich fortschrittliches Verfahren verteilter Intrusion Detection Systeme eingehen, das erst kürzlich auf Grundlage einer frühen, prototypischen Implementierung vorgestellt wurde und somit besonders aktuell ist. Wie DOMINO implementiert auch LarSID eine hoch entwickelte, leistungsfähige und robuste Peer-to-Peer Architektur auf Basis verteilter Hashtabellen.

Wiederum über ein dynamisches Overlay-Netz wird eine übergreifende Plattform etabliert, um die aus einzelnen auffälligen Beobachtungen aggregierten und bereits rudimentär vorverarbeiteten sicherheitsrelevanten Ereignisse effizient zwischen einer Vielzahl von organisationsübergreifenden Systemen auszutauschen, was in besonderem Maße gerade auch die Erkennung von großflächig angelegten Angriffen schon in einem frühen Stadium unterstützen soll.

Im Unterschied zu DOMINO ist LarSID allerdings durchgängig als dienstorientiertes Framework konzipiert, wodurch vor allem die Wartbarkeit und die Verbreitung erleichtert wird. Die spezifische Dienstschicht wurde von der Schicht des verteilten Hashings getrennt, wobei für letztere explizit auch unterschiedliche Ausprägungen Anwendung finden können. Beispielhaft wurde LarSID mittels des „OpenDHT“ [Opea] realisiert, einem bekannten, frei nutzbaren Service für verteilte Hashtabellen.

Die grundlegende Architektur von LarSID sieht ein flaches Publish/Subscribe-Modell vor. Sämtliche im Overlay-Netz organisierten Komponenten tauschen in regelmäßigen Abständen die in ihrem Bereich lokal erfassten und potenziell sicherheitskritischen Ereignisse mit den übrigen Teilnehmern aus. Bestätigt sich auf diese Weise der Verdacht eines Angriffs, ergeht eine entsprechende Rückmeldung an die betreffenden Systeme. Die Kommunikation für den Austausch von Meldungen erfolgt jeweils anonym.

Alle Systeme haben demzufolge gleichzeitig zwei Rollen inne. Zum einen fällt es in ihren Aufgabenbereich, relevante Beobachtungen zu erfassen, zu aggregieren sowie sie anschließend zur kooperativen Analyse weiterzuleiten. Gegebenenfalls erhalten sie außerdem Benachrichtigungen über bestätigte Angriffe. Auf der anderen Seite sind sie ebenfalls verantwortlich dafür, von anderen Systemen erhaltene Berichte zu korrelieren und im Falle eines vermeintlich erkannten Angriffs sodann selbst Benachrichtigungen für die Betroffenen zu generieren.

Während der Entwicklung der vorgeschlagenen Lösung wurde ein besonderes Augenmerk auf eine Steigerung der Erkennungsgenauigkeit gelegt. Auf Basis der empirischen Analyse eines umfangreichen, der Realität möglichst nahekommenden Musterdatensatzes (dem DShield Datensatz [DSh]), der von mehr als 1600 Firewalladministratoren weltweit gesammelt wurde, konnten wichtige Parameter der vorgeschlagenen Lösung maßgeblich optimiert werden.

Insgesamt bleibt festzuhalten, dass der Ansatz der verteilten Intrusion Detection Systeme verglichen mit der klassischen Einbruchserkennung einen bedeutenden Fortschritt darstellt. Dennoch werden neue, gridspezifische Lösungen benötigt, die nunmehr auch die individuellen Anforderungen im

Umfeld von Grids explizit berücksichtigen. Der aktuelle Stand der Forschung an solchen gridspezifischen Intrusion Detection Systemen ist Inhalt des folgenden Abschnitts.

3.2 Neue, gridspezifische Ansätze

Die in den letzten Abschnitten untersuchten verteilten Intrusion Detection Systeme verbessern bedeutend die Fähigkeiten von Systemen zur Angriffserkennung, indem die Informationen mehrerer administrativer Domänen und sogar mehrerer Organisationen zur Gewinnung eines umfassenderen Bildes der Sicherheitslage zusammengeführt und gemeinsam ausgewertet werden. Wie bereits eingehend diskutiert, stellen Grid Umgebungen daneben allerdings weitergehende, besondere Anforderungen an das Sicherheitsmanagement, beispielsweise bedingt durch die neuartige Zusammenarbeit in virtuellen Organisationen und die damit verknüpfte Verwendung spezieller Grid-Middlewares für den koordinierten Zugriff auf verteilte Ressourcen.

Seit etwa dem Jahr 2002 rückte daher die Arbeit an angepassten Intrusion Detection Systemen allmählich in den Fokus der wissenschaftlichen Forschung, die die diversen Limitierungen vorhandener Verfahren in Bezug auf die Nutzung im Grid überkommen möchte. Obwohl der Bereich offensichtlich vielschichtige Fragestellungen und ein großes Betätigungsfeld für Verbesserungen offeriert, gibt es bis heute allerdings nur eine überschaubare Anzahl an darauf bezogenen Veröffentlichungen, die zudem meist knapp gehalten sind und wenig ausgereift erscheinen.

Die anschließenden Abschnitte analysieren nunmehr den aktuellen Entwicklungsstand auf dem Gebiet gridspezifischer Intrusion Detection Systeme. Dazu werden die wichtigeren Konzepte jeweils separat dargestellt, wobei auch die individuellen Schwachpunkte kurz aufgezeigt werden sollen. Die zu erwartende zunehmende Verbreitung von Grids sollte zukünftig eine rege Weiterentwicklung erhoffen lassen.

3.2.1 Grid-based Intrusion Detection System (GIDS)

Eines der ersten bekannten und etwas detaillierteren Konzepte einer Einbruchserkennung spezifisch für Grids wurde an der Universität von Penang (Malaysia) als „Grid-based Intrusion Detection System“ (GIDS) [CS03] vorgestellt. Im Unterschied zu einigen früheren Veröffentlichungen (vgl. unter anderem [LPO+03, TBK03, LOR+03]), wo die Notwendigkeit gridspezifischer Intrusion Detection nur allgemein diskutiert wurde, fokussiert die Arbeit erstmals auch auf eine tatsächliche Lösung der vielfältigen Probleme. Bestandteil der Analyse sind insbesondere auch einige Beispielszenarien, anhand derer sich die Einsatzmöglichkeiten von Intrusion Detection Systemen im Umfeld des Grid Computing näher untersuchen lassen.

Eine entscheidende Neuerung des Ansatzes stellt die ausdrückliche Berücksichtigung einzelner wichtiger Grid-Spezifika dar. Insbesondere wird dabei das bedeutende Konzept der virtuellen Organisationen (VOs) (vgl. Abschnitt 2.3.2.2) aufgegriffen. Den Autoren folgend, integriert sich das GIDS selbst als virtuelle Organisation in die Grid-Umgebung (vgl. Abbildung 3.5). Die vorgebrachten Ideen sind allerdings allesamt nur theoretischer Art, eine wenigstens prototypische Implementierung wurde im weiteren nicht mehr publiziert.

Das vorgeschlagene GIDS sieht eine einfache, zentralisierte Architektur vor, die im wesentlichen auf den einzelnen Komponenten Agent, Server und Manager beruht. Agenten befinden sich als leichtgewichtige Hintergrundprozesse VO-übergreifend auf den im Grid verbundenen und potenziell heterogenen Ressourcen. Zu ihrem Verantwortungsbereich zählen lediglich das Sammeln von sicherheitsrelevanten Daten des lokalen Hosts, deren Vorverarbeitung im Sinne einer wirksamen Kompression

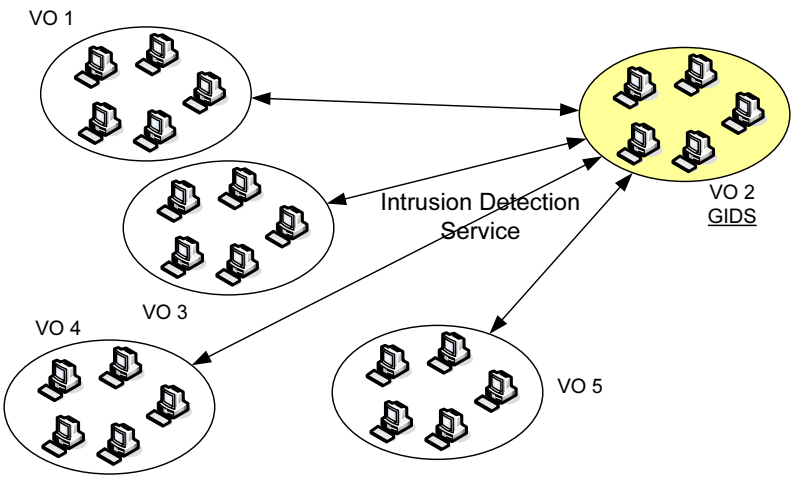

Abbildung 3.5: Integration des GIDS als virtuelle Organisation (nach [CS03], Abbildung 2)

und Verschlüsselung sowie die sichere Übermittlung dieser Informationen an den übergeordneten Server. Ausschließlich in der zentralen Komponente des Servers wird dann die gesamte Analyselogik gebündelt. Im Manager sind schließlich alle Verwaltungsdienste im Zusammenhang mit dem Betrieb und der Überwachung des Systems zusammengefasst.

Der zentralisierte Aufbau weist freilich die bereits in Abschnitt 3.1.1 erläuterten Defizite einfacher, verteilter Intrusion Detection Systeme unter anderem hinsichtlich seiner Skalierbarkeit und Verletzbarkeit auf, was die Praxistauglichkeit demzufolge über die Maßen einschränkt. Das Konzept streift nur rudimentär eine Lastverteilung zwischen mehreren für die Analyse zuständigen Servern, ohne darauf jedoch in der erforderlichen Tiefe einzugehen. Kennzeichnend für den gesamten Entwurf ist es, dass keinerlei Modellbildung von potenziellen Grid-Bedrohungen zugrunde liegt oder im Rahmen des Projekts initiiert wurde. Dieses mangelnde Verständnis von gridspezifischen Vorfällen engt das GIDS hauptsächlich auf wenige offensichtliche Angriffe ein, wie die beispielhaft erwähnte missbräuchliche Nutzung von Ressourcen des Grids.

3.2.2 Grid Intrusion Detection Architecture (GIDA)

Auf Grundlage der früheren Versuche des GIDS befasste man sich in der Folge auch an der Universität von Kairo (Ägypten) mit den gewichtigen Herausforderungen von Intrusion Detection Systemen für Grids. Es wurde das vielbeachtete Konzept einer neuen, offenen und flexibleren „Grid Intrusion Detection Architecture“ (GIDA) [TAWTAS05b, TAWTAS05a] oberhalb der „Grid Security Infrastructure“ (GSI) (vgl. Abschnitt 2.3.4.3) entwickelt, das die Einbruchserkennung nicht nur ins Grid integriert, sondern auch stärker vom spezifischen Umfeld profitieren lassen soll. Eine angepasste Grid-Simulation basierend auf „GridSim“ [MBA02] wurde verwendet, um die Ergebnisse zu verifizieren und das Verfahren weiter zu optimieren.

Um eine ausreichende Skalierbarkeit sicherzustellen, verzichtet GIDA auf eine zentralisierte Organisation der Komponenten und sieht stattdessen explizit auch eine verteilte Analyse aller sicherheitsrelevanten Ereignisse vor. Der Aufbau der GIDA umfasst zwei Hauptbestandteile, „Intrusion Detection Agenten“ (IDA) und „Intrusion Detection Server“ (IDS) (vgl. Abbildung 3.6). Jede administrative Domäne weist einen Agenten auf, der die relevanten Daten einsammelt und weiterleitet. Der Austausch dieser Informationen gleich mit mehreren übergeordneten Servern ist hierbei als ein einfacher Replikationsmechanismus angedacht, damit der Ausfall eines einzelnen Servers nicht den

Ausfall des gesamten Intrusion Detection Systems nach sich zieht. Zu Fragen etwa der Kompression oder Sicherung nehmen die Autoren nicht weiter Stellung.

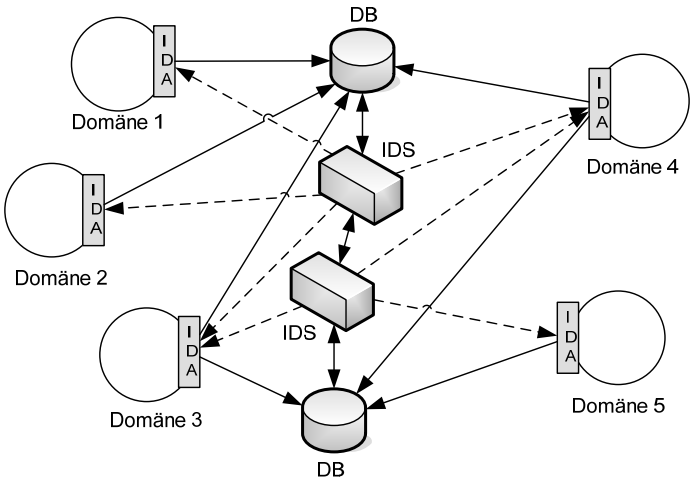

Abbildung 3.6: Aufbau der GIDA (nach [TAWTAS05b], Abbildung 1)

In den Verantwortungsbereich der Server fällt neben der Aggregation und Korrelation der gemeldeten Ereignisse seines Verwaltungsbereiches sowie etwaigen Warnungen an von Angriffen betroffenen administrativen Domänen diesmal auch die Kooperation mit den übrigen Servern des Grids, wobei dies analog des Peer-to-Peer Ansatzes (P2P) (vgl. Abschnitt 3.1.3) geschehen soll. Die Autoren setzen hierfür ein quasi ideales System voraus, weshalb die Herausforderungen eines solchen Verfahrens folglich nicht genauer behandelt werden.

Wenngleich die Arbeit auch heterogene Ressourcen fast beiläufig erwähnt, wurden die angesprochenen Tests allesamt in einer homogenen Umgebung durchgeführt. Die Anwendung auf ein heterogenes Szenario wurde als zukünftige Weiterentwicklung offen gelassen. Es zeigte sich unter anderem, dass eine Vergrößerung der Anzahl der Server aufgrund der temporär jeweils stärker eingeschränkten Sicht zu mehr falsch positiven Meldungen führte, aber auch zu weniger falsch negativen. Vergrößert man die Menge der verfügbaren Ressourcen, ging damit eine Verminderung falsch positiver Meldungen einher.

Durch die fortschrittliche Architektur kooperierender Server kann dem Problem der begrenzten Skalierbarkeit wirkungsvoller begegnet werden. Im Vergleich zum GIDS ist die vorgeschlagene Lösung aber recht allgemein gehalten und fokussiert wenig auf die kritischen Eigenheiten der Grid-Umgebung. Speziell die für Grids besonderen virtuellen Organisationen und die damit verbundenen Anforderungen an das Management einer solchen Anwendung lässt die Arbeit in weiten Teilen aus. Wie im Falle des GIDS wurde gleichfalls für GIDA kein Modell möglicher Angriffsszenarien entwickelt, so dass nur grundlegende Typen in Betracht gezogen wurden. Das System soll immerhin zukünftig die Herleitung einer betreffenden Signaturensammlung erleichtern.

3.2.3 Performance-based Grid Intrusion Detection System (PGIDS)

Mit dem „Performance-based Grid Intrusion Detection System“ (PGIDS) [LLLY05b, LLLY05a] wurde eine in einigen nennenswerten Punkten abweichende Vorgehensweise empfohlen, bei der die Knoten des Grids selbst unter Anwendung eines fortgeschrittenen Verfahrens zur Lastverteilung für die Angriffsanalyse eingesetzt werden. Anders als manch ähnliche Veröffentlichung wurden die Ergebnisse durch einfache experimentelle Versuche bestätigt.

Die Arbeit zielt hauptsächlich auf die Erkundung von groß angelegten Angriffen auf die zu schützenden Netze ab, wobei dabei die vergleichsweise großen Grid-Kapazitäten auch die Verarbeitung von starkem Netzverkehr zulassen. Das Verfahren beruht auf der Bereitstellung eines autonomen PGIDS je einzubeziehender administrativer Domäne. Die Architektur sieht hierfür mehrere in das Umfeld des Grids integrierte Komponenten vor: mehrfache Dispatcher, einen Scheduler und die analysierenden Knoten (*Detection Nodes*), sowie eine unterstützende Datenbank (*Block List Database*).

Die einzelnen Subnetze innerhalb einer administrativen Domäne verfügen jeweils über einen Dispatcher, der regelmäßig den aufgelaufenen Netzverkehr unter Zuhilfenahme grideigener Protokolle an einen gerade verfügbaren Knoten zur eigentlichen Durchführung der Einbruchserkennung übergibt. Die Auswahl desselben aus einer womöglich sehr großen Menge erledigt der Scheduler. Indem hierbei die aktuellen Systemzustände aktiv miteinbezogen werden, kann die Geschwindigkeit der Verarbeitung drastisch erhöht und die Berechnungslast effizient aufgespalten werden. Die assoziierte Datenbank kommt ausschließlich der Dokumentation erkannter Angriffe zugute, mit deren Hilfe sich beispielsweise Firewalls verbessern lassen.

Alles in allem spart das Konzept neben der Frage der domänenübergreifenden Kooperation jedoch viele gridspezifische Herausforderungen an Systeme zur Einbruchserkennung aus. Zentrale Bausteine können bei sehr großen Topologien zu einem Flaschenhals werden. Die vordergründige Ausrichtung auf Netzangriffe verstellt den Blick auf andere mindestens ebenso kritische Ausprägungen möglicher Bedrohungen, die im weiteren Verlauf nicht aufgearbeitet werden. Auch wenn das System bereits weitgehend aus der Struktur von Grids Nutzen schlägt, verzichtet es vollständig auf eine tiefergreifende Untersuchung genereller Anforderungen und damit verbundener Aspekte von Anwendungen des Sicherheitsmanagements.

3.2.4 Integrated Grid-based Intrusion Detection System

Ein weiterer interessanter Beitrag in Bezug auf integrierte gridspezifische Intrusion Detection Systeme stammt von der Universität von Santa Catarina (Brasilien) [SNKW06, SRKW06]. Die Arbeit untersucht zunächst allgemein mögliche Angriffsszenarien betreffend Grids sowie wichtige Anforderungen an Systeme zur Einbruchserkennung. Im Anschluss wird ein konkreter Lösungsvorschlag gegeben und gegenüber bestehenden Ansätzen abgegrenzt.

Im Gegensatz zu den vorangegangenen Konzepten, die eine genauere Untersuchung und Klassifizierung von Angriffen auf Grids vor dem Hintergrund der offensichtlichen Komplexität des Themas ignorierten, wird diesmal zumindest ansatzweise darauf eingegangen. Es werden diesbezüglich vier Klassen unterschieden, die zugleich auch als Fundament für die weitere Analyse herangezogen werden: unerlaubte Zugriffe, missbräuchliche Nutzung, Exploits und host- oder netzspezifische Angriffe. Als generelle Unterschiede zwischen Angriffen auf Grids und solchen auf herkömmliche verteilte Systeme werden die zu erwartende höhere Geschwindigkeit und das aller Voraussicht nach größere Schadenspotenzial angesehen.

Der anfänglich vereinfachte Entwurf umfasst generell ein übergeordnetes Intrusion Detection System (GIDS), das die Funktionalitäten von herkömmlichen hostspezifischen Systemen (HIDS) auf den einzelnen Knoten sowie netzspezifischen Systemen (NIDS) per administrativer Domäne zum Zwecke der gridbasierten Einbruchserkennung zusammenführt (vgl. Abbildung 3.7). Auf diese Weise lassen sich übergreifend sicherheitskritische Vorfälle ableiten, um dann gegebenenfalls Alarmmeldungen generieren zu können. Für die interoperable Kommunikation zwischen inkompatiblen Intrusion Detection Systemen kommt wiederum ein auf dem „Intrusion Detection Message Exchange Format“ (vgl. Abschnitt 2.2.5.4) basierendes Protokoll zum Einsatz, wodurch aufwendige Reimplementierungen vermieden werden und stattdessen vorhandene Lösungen weiterhin Verwendung finden können.

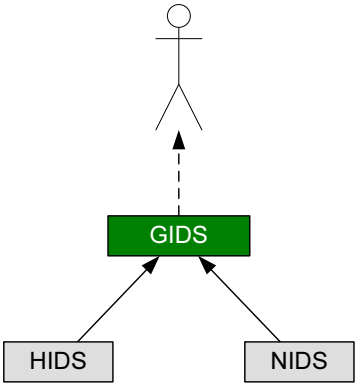

Abbildung 3.7: Allgemeiner Aufbau [SNKW06]

Für die Umsetzung eines solchen Schemas wurde eine abstrakte, dreigeteilte Architektur entwickelt, die sich auf Agenten, Analysekomponenten und eine Steuereinheit stützt. Agenten auf den einzelnen Knoten greifen die sicherheitsrelevanten Informationen von den individuellen, einzubindenden Intrusion Detection Systemen ab und speichern sie in vorhandenen Grid-Datenbanken. Bei jedem Zugriff eines Anwenders auf das Grid erfolgt eine durch die Steuereinheit initiierte Überprüfung gegen das in der Datenbank gespeicherte Nutzerprofil. Gegebenenfalls werden Aufträge für eine weitergehende Analyse an Knoten des Grids mit freien Berechnungskapazitäten vergeben, die ihrerseits mit den Datenbanken kommunizieren und die Profile aktualisieren. Die Analyse ist ebenfalls für die Aggregation und Korrelation gespeicherter sicherheitsrelevanter Ereignisse auch untereinander verantwortlich, um so verteilte Angriffe identifizieren zu können.

Verglichen mit dem GIDS und GIDA umfasst die Darstellung immerhin eine rudimentäre Analyse kritischer Angriffsszenarien. Ob diese einfache Form der Auseinandersetzung allerdings ausreicht, um ein gridspezifisches Intrusion Detection System theoretisch zu fundieren, muss der weitere Verlauf der Forschung erst zeigen. Auf jeden Fall muss die praktische Nutzbarkeit und besonders die versprochene hohe Skalierbarkeit ohne eine wenigstens prototypische Implementierung bezweifelt werden. Auch wenn der Ansatz durch die Verwendung von Ressourcen des Grids vom Umfeld profitieren kann, bleiben wesentliche Herausforderungen, wie etwa Aspekte betreffend das organisationsübergreifende Management gridspezifischer Einbruchserkennung, noch unangetastet. Die Fähigkeiten der herangezogenen herkömmlichen Intrusion Detection Systeme sind in Bezug auf gridtypische Angriffsszenarien in hohem Maße beschränkt, da sie beispielsweise nur lokale, aber keine Gridnutzer kennen. Generell können Angriffe auf Grids einige Besonderheiten aufweisen, die vom Konzept nicht berücksichtigt werden und so potenziell zu hohen Fehlerraten führen können.

3.2.5 Weitere Projekte

Neben den in den vorausgegangen Abschnitten aufgeführten bedeutungsvollen Ansätzen gibt es noch eine überschaubare Anzahl weiterer mehr oder weniger ausgereifter Vorschläge, wie dem Problem der Einbruchserkennung spezifisch für Grids begegnet werden kann. Da sich im Rahmen dieser Arbeit nicht sämtliche Konzepte ausführlich darstellen lassen, soll dieser Abschnitt den angestrebten Überblick zu gridbasierten Intrusion Detection Systemen mit einer kurzen Darstellung zweier recht aktueller chinesischer Beiträge abschließen.

3.2.5.1 Grid-specific Host-based Intrusion Detection System (GHIDS)

Das „Grid-specific Host-based Intrusion Detection System“ (GHIDS) [FDL+06] fokussiert hauptsächlich auf die Weiterentwicklung herkömmlicher hostbasierter Intrusion Detection Systeme im Hinblick auf die Analyse einiger gridtypischer Angriffe. Berücksichtigten solche Systeme bislang nur lokale Benutzer, so stellt das GHIDS Zugriffe auf Ressourcen nun in den Kontext des Gridnutzers. Es wird angenommen, dass die so gewonnenen umfangreicheren Informationen zu potenziellen Angriffsversuchen dann von einem übergeordneten Intrusion Detection System gridübergreifend zum Zwecke einer weitergehenden Angriffsanalyse Verwendung finden, wobei hierzu keine weiteren Aussagen getroffen werden. Die Funktionsfähigkeit der Lösung wurde anhand eines einfachen, reduzierten Prototyps gezeigt.

Um auch neuartige Angriffe erkennen zu können, stützt sich das GHIDS auf die sogenannte „Bottleneck Verification“ (BV), die einige wichtige sicherheitsrelevante Ereignisse im Vergleich zu etwa der Anomalieanalyse (vgl. Abschnitt 2.2.4.2) besonders effizient analysieren soll und zudem die Performanz des zu überwachenden Hosts nur sehr gering beeinträchtigt. BV-basierte Intrusion Detection Systeme erreichen für wichtige Angriffsszenarien oftmals niedrige Fehler- und hohe Erkennungsraten. Im wesentlichen stützt sich das Verfahren auf die Beobachtung von Überschreitungen der individuellen Rechte der Systembenutzer.

Das Konzept behandelt nur einen Ausschnitt der gesamten Problemstellung und lässt viele Fragen für Systeme zur Einbruchserkennung im Grid offen, indem ausschließlich die hostbasierte Analyse für das Grid fortentwickelt wird. Gerade auch zur aggregierten und korrelierten Auswertung der sicherheitsrelevanten Meldungen vieler derartiger Systeme im Umfeld hochgradig verteilter, domänenübergreifender Grids werden keine Antworten gegeben. Insbesondere stellt ein zentrales, übergeordnetes System potenziell einen schwerwiegenden Flaschenhals dar.

3.2.5.2 Self-adaptive Intrusion Detection System for Computational Grid

Die bisher besprochenen Ansätze weisen oftmals eine vergleichsweise statische Anordnung der Komponenten auf, die sich gegenüber Veränderungen wenig robust gibt. Zuletzt soll mit dem „Self-adaptive Intrusion Detection System for Computational Grid“ [NLSX07] noch eine Arbeit erwähnt werden, die vor dem Hintergrund hochgradig dynamischer Gridstrukturen auf einem neuartigen hierarchisch organisierten, immunen System basierend auf einer Menge von autonomen Agenten beruht. Zur Überprüfung der Ergebnisse wurden einige Experimente in einem einfachen, simulierten Gridumfeld durchgeführt.

Immune Systeme (vgl. [GE06, Li05]) können sich an Änderungen der Systemumgebung selbst anpassen, wie dies gerade in großflächig verteilten Gridszenarien häufig auftreten kann. Sie sind zuletzt mehr und mehr in den Fokus wissenschaftlicher Veröffentlichungen gerade im Sicherheitsbereich gerückt. Von ihren interessanten Eigenschaften können nicht zuletzt besonders auch Intrusion Detection Systeme profitieren. Leichtgewichtige mobile und unabhängige Agenten (vgl. [HFAAA05, AIA05]) adaptieren sich dabei auf einfache Weise sowohl an individuelle Veränderungen der Infrastruktur als auch an neue Anforderungen, wodurch eine effektive Angriffserkennung im Grid wirkungsvoll unterstützt werden kann. Das aufgezeigte Verfahren bezieht dabei auch explizit existierende Protokolle und Sicherheitsmechanismen des Grids mitein, deren Schwachstellen so ebenfalls überwacht werden können.

Durch die Integration mobiler Agenten und die hierarchische Architektur kann die Skalierbarkeit grundsätzlich verbessert werden. Trotzdem geht die Arbeit kaum auf die vielfältigen Probleme der

organisationsübergreifenden Zusammenarbeit spezifisch für das Grid ein, wodurch die Praxistauglichkeit weithin infrage gestellt werden muss.

3.3 Zusammenfassung

Inhalt des Kapitels ist ein Überblick zum aktuellen Entwicklungsstand gridbasierter Intrusion Detection Systeme.

Hierfür werden anfangs in Abschnitt 3.1 zumindest die wichtigsten Konzepte der den gridspezifischen Lösungen vorausgehenden verteilten Intrusion Detection analysiert. Im Vergleich zu klassischen Mechanismen der Einbruchserkennung stellen verteilte Verfahren einen bedeutenden Fortschritt besonders hinsichtlich des domänen- und sogar organisationsübergreifenden Einsatzes dar. Indem nunmehr viele Systeme an der Datenkollektion partizipieren, die nicht unbedingt einer administrativen Domäne angehören und nicht zwingend unter eigener Kontrolle stehen müssen, verbessert sich direkt das Verständnis von sicherheitsrelevanten Aktivitäten. Um das Themenfeld umfassend und vollständig abzubilden, unterscheidet die vorliegende Arbeit mehrere Phasen von den Anfängen bis zum gegenwärtigen Zeitpunkt, wobei jeweils einzelne typische Vertreter exemplarisch untersucht werden. Auf die anfänglich einfacheren zentralisierten Verfahren folgten rasch fortschrittlichere hierarchisch organisierte Konzepte, die vor allem die Skalierbarkeit durch eine aufgeteilte, dezentrale Analyse steigern. Aktuelle Projekte zielen vermehrt auf einen robusten Einsatz in sehr großen Umgebungen.

Obgleich die zuvor beschriebenen verteilten Intrusion Detection Systeme die Fähigkeiten von Systemen zur Angriffserkennung bereits entscheidend voranbringen, werden dennoch neue Lösungen benötigt, die die diversen Limitierungen vorhandener Verfahren in Bezug auf die Nutzung speziell im Grid überkommen. Trotz offensichtlich vielschichtiger Fragestellungen, gibt es bislang nur eine überschaubare Zahl an diesbezüglichen Veröffentlichungen. Abschnitt 3.2 behandelt davon die wichtigeren Forschungsprojekte neuer gridbasierter Intrusion Detection Systeme, wobei vor allem auch deren jeweilige Schwachstellen kurz aufgezeigt werden. Dabei ist für alle Entwürfe kennzeichnend, dass jeweils nur auf einzelne Teilaspekte der gesamten Problematik im Grid isoliert fokussiert wird, ohne vorher im Sinne eines wirklich geplanten Entwicklungsprozesses das erforderliche breite Fundament und hierbei besonders die grundsätzlichen Anforderungen im Umfeld von Grids in angemessener Weise diskutiert zu haben. Die Übertragbarkeit auf reale Szenarien muss daher weithin bezweifelt werden.

KAPITEL VIER

Analyse zur Gewinnung von wichtigen generischen Anforderungen an gridbasierte Intrusion Detection

Inhalt des Kapitels

4.1 Vorgehensweise der Anforderungsanalyse 50
4.2 Beschreibung der Aktoren 50
4.3 Beschreibung der Anwendungsfälle 52
4.3.1 Organisationsübergreifende Zusammenarbeit 52
4.3.2 Intrusion Detection in Grids 55
4.3.3 Integration in übrige Systeme 57
4.3.4 Ergänzende, grundlegende Anwendungsfälle 57
4.3.5 Übersicht über die Anforderungen 60
4.4 Abgeleitete generische Anforderungen 60
4.5 Zusammenfassung 71

Nachdem in den vorausgegangenen Kapiteln abgesehen von wichtigen Grundlagen auch bereits der aktuelle Stand der Forschung besprochen wurde, soll dieses Kapitel nun, wie es der Titel der Arbeit ankündigt, bezüglich eines soliden, planvollen Prozesses im Rahmen der Softwareentwicklung den Kern der Arbeit umfassen. Hierfür werden initial die generischen Anforderungen an Intrusion Detection Systeme im Umfeld von Grids in Form einer strukturierten Analyse ermittelt.

Abschnitt 4.1 gibt vorweg einen allgemeinen Überblick über die im Hinblick auf die Anforderungsanalyse gewählte Vorgehensweise. Neben konzeptuellen Erläuterungen soll hierbei besonders auch die Zielsetzung der Anforderungsanalyse unterstrichen werden.

Als Ausgangspunkt für die differenzierte Auseinandersetzung befasst sich Abschnitt 4.2 zunächst mit den einzelnen Aktoren, die mit gridbasierten Intrusion Detection Systemen interagieren. Hierauf aufbauend arbeitet Abschnitt 4.3 relevante Anwendungsfälle heraus, um daraus die jeweils maßgeblichen, unterschiedlichen Anforderungen an die Einbruchserkennung im Grid fundiert ableiten zu können. In der Folge widmet sich Abschnitt 4.4 schließlich ausführlich sämtlichen gefundenen Anforderungen, wobei diese jeweils separat erklärt und mit einer Erläuterung ihrer Relevanz versehen werden.

4.1 Vorgehensweise der Anforderungsanalyse

In Abschnitt 2.4 wurden bereits die vielfältigen Herausforderungen angesprochen, denen Einbruchserkennungssysteme in heterogenen Grid-Umgebungen generell gerecht werden müssen. Die anzustrebende übergreifende Lösung muss sich dazu bestmöglich in ein anspruchsvolles, hoch dynamisches Umfeld integrieren können.

Um nun die generischen Anforderungen, die an Einbruchserkennungssysteme spezifisch in Grid-Umgebungen zu stellen sind, systematisch und substanziiert herauszuarbeiten, ist ein wohl überlegtes Vorgehen unabdingbare Voraussetzung. Zu einem früheren Zeitpunkt wurden am hiesigen „Leibniz-Rechenzentrum“ bereits innerhalb des „D-Grid“ (vgl. Abschnitt 2.3.5) als Teil einer umfassenderen Studie die Anforderungen an ein integratives Monitoring erhoben [BOvdgFGR06]. Für die prinzipiell verwandte Analyse von Anforderungen an Intrusion Detection Systeme im Grid wird demnach ein vergleichbarer Ansatz gewählt.

Die Anforderungen an die zu entwickelnde Lösung stammen aus dem charakteristischen Anwendungsbereich und werden infolgedessen aus den Erfordernissen der dort relevanten technischen Prozesse und möglicherweise vorhandener Workflows abgeleitet. Nach der kurzen, einleitenden Beschreibung der beteiligten Aktoren liegt also vor allem auf der durchdachten, zielgerichteten Formulierung der Anwendungsfälle das Hauptaugenmerk der Arbeit, um hieraus vordergründig die unterschiedlichen Anforderungen gewinnen zu können. Jeder Anwendungsfall beschreibt möglichst praxisnah jeweils einen signifikanten Gesichtspunkt, dem sich sodann die einzelnen Anforderungen zuordnen lassen.

Im Zuge der Analyse soll ausdrücklich ein möglichst allgemeiner Blickwinkel eingenommen werden, um die Applikabilität des Ansatzes bezogen auf das breite Feld an anzunehmenden Zielumgebungen gewährleisten zu können. Dem hierfür notwendigen Abstraktionsgrad wird durch zwar praxisnah, aber gleichwohl übergreifend instantiierbar modellierte Anwendungsfälle vollständig genüge getan. An die Anforderungen werden aufgrund der umfassenden Zielsetzung generell hohe Ansprüche gestellt. Im wesentlichen umfasst dies:

- Möglichst vollständige Abdeckung aller Anforderungen durch strukturierte Auswahl relevanter Anwendungsfälle
- Feststellung „atomarer“ Anforderungen
- Hohe Präzision und Abgrenzung der Anforderungen gegeneinander
- Konsistente, das heißt widerspruchsfreie und verständliche Definition der Anforderungen
- Einheitliche, formalisierte Dokumentation

Aufgrund der dem Vorgehen insoweit inneliegenden hohen Allgemeingültigkeit wurde folgerichtig auf eine Trennung der funktionalen und nicht-funktionalen Anforderungen verzichtet.

4.2 Beschreibung der Aktoren

Wichtiger Bestandteil einer jeden Anforderungsanalyse muss zunächst die hinreichend erschöpfende Identifikation aller beteiligten Aktoren sein.

Aktoren treten im Rahmen einer solchen Analyse nicht in ihrer Eigenschaft als Person (beziehungsweise Ereignis oder Prozess) auf, sondern schlüpfen in eine Rolle, die in der Regel eine bestimmte Funktion innerhalb des betrachteten Szenarios repräsentiert. Sie kann von verschiedenen Aktoren

gleichzeitig oder nacheinander eingenommen werden. In das anzustrebende gridbasierte Intrusion Detection System sind dementsprechend mehrere Aktoren in unterschiedlichen Rollen eingebunden. (vgl. [ER02])

Die Komplexität einer Einbruchserkennung im Grid führt zu einer Vielzahl an abzugrenzenden Aktoren. Es lassen sich jedoch primär die folgenden Aktoren identifizieren:

Ressourcenanbieter. Ressourcenanbieter sind direkt an einer gridbasierten Einbruchserkennung beteiligt. Sie stellen Kontingente ihrer lokalen Ressourcen und Dienste für eine Nutzung durch das Grid zur Verfügung, die sodann auch von potenziellen Angriffen bedroht sind.
Der Erfolg und sicherheitstechnische Nutzen eines gridübergreifenden Intrusion Detection Systems hängt in hohem Maße davon ab, ob und in welcher Qualität relevante Informationen von den einzelnen eingebundenen Partnern einbezogen werden können. Dabei erwarten Ressourcenanbieter gleichfalls eine individuelle, für sie angepasste Berichterstattung.

Lokaler IDS-Administrator. Für die Erbringung einer gridübergreifenden Einbruchserkennung ist vor allem die Zusammenarbeit mit Administratoren lokaler Intrusion Detection Systeme unerlässlich.
Lokale IDS-Administratoren erbringen diverse Aufgaben hinsichtlich des Betriebs einer wirkungsvollen Angriffserkennung. Zu den hierbei wesentlichen Aufgaben zählen hauptsächlich die Kalibrierung lokaler Intrusion Detection Systeme, Anpassungen bei eventuellen Veränderungen der Einsatzumgebung, die regelmäßige Aktualisierung von Signaturen oder von Systemen selbst und auch die Auswertung und Verfolgung von den im Rahmen der Einbruchserkennung gemeldeten Ereignissen und Alarmen. Daneben prüfen sie regelmäßig die Funktionsfähigkeit der Komponenten.

Grid Operations Center. In ausgedehnten Grid-Umgebungen wird dem zentralen „Grid Operations Center“ im Hinblick auf den Betrieb einer organisationsübergreifenden Erkennung von Angriffen offenkundig eine tragende Rolle zuteil, wobei neben einigen grundsätzlichen Managementaspekten besonders auch zusätzliche unterstützende Aufgaben von besonderer Relevanz sind.
Da sich typische Grid-Verbünde heute in vielen Fällen noch in der Entwicklung befinden, sind die genauen Strukturen und die Aufgaben häufig noch nicht vollständig ausgearbeitet und dokumentiert (vgl. unter anderem [BDE+07]). Prinzipiell gehört zum Tätigkeitsbereich vor allem die fortwährende Überwachung aller wichtigen Ressourcen und Dienste des Grids, speziell also auch eines gridbasierten Intrusion Detection Systems, um dadurch dauerhaft eine hohe Verfügbarkeit gewährleisten zu können. Kommt es zu Ausfällen oder Fehlern, koordiniert und kontrolliert das „Grid Operations Center“ dementsprechend ebenso die rasche Wiederherstellung betroffener Komponenten. Zusätzlich kann eine zeitlich durchgehende Unterstützung von Nutzern, virtuellen Organisationen und Ressourcenanbietern bei anstehenden Problemen und Fragen geleistet werden.
Hierfür bedarf es entsprechend einer permanenten Erreichbarkeit wechselnder kompetenter, diensthabender Grid-Operatoren.

Grid IDS-Administrator. Die Abgrenzung von Grid IDS-Administratoren gegenüber lokalen IDS-Administratoren ist nicht unbedingt offensichtlich. Die Trennung beider Aktoren ist jedoch für die weitere Auseinandersetzung mit der Anforderungsanalyse fundamental.
Die Rollenausprägung des Grid IDS-Administrators ist deutlich von der des lokalen IDS-Administrators zu unterscheiden. Im Falle des lokalen IDS-Administrators stehen die etwaig vorhandenen organisationsinternen, lokalen Intrusion Detection Systeme im Mittelpunkt, wohingegen die Zuständigkeit von Grid IDS-Administratoren in der umfassenden, gridübergreifenden Einbruchserkennung liegt. Insofern werden von Grid IDS-Administratoren verglichen

mit denen lokaler Umgebungen ähnliche Aufgaben übernommen, diesmal nur auf einer höheren Stufe.

Virtuelle Organisation. Neben den vorgenannten Aktoren sind unzweifelhaft ebenfalls virtuelle Organisationen, beziehungsweise deren jeweilige Mitglieder in verschiedenen Rollenausprägungen explizit an einer gridweiten Lösung zur Angriffserkennung beteiligt.
Sie erwarten zunächst eine angepasste Berichterstattung bezüglich ihres begrenzten Aktionsbereichs, also in puncto ihrer VO-internen Ressourcen und Dienste. Des weiteren interagieren sie aber auch mit dem System, um flexibel etwaige Anpassungen vorzunehmen; im gegebenen Fall kommen mitunter administrative Tätigkeiten hinzu.

Einzelner Nutzer. Prinzipiell sind von typischen Grid-Verbünden ebenfalls einzelne Nutzer zu beachten, die keiner der anerkannten virtuellen Organisationen angehören. Da sich einzelne Nutzer aber gegebenenfalls auf virtuelle Organisationen mit entsprechend nur einem Mitglied abbilden lassen, gilt das auf virtuelle Organisationen Bezogene weitgehend analog.

Neben den aufgeführten Aktoren sind weitere ergänzende Aktoren, vor allem technischer Art von Interesse. Dazu gehören beispielshalber die integrierenden Middlewares oder diverse Managementsysteme (etwa zur Steuerung und Überwachung des Ressourceneinsatzes).

4.3 Beschreibung der Anwendungsfälle

Nach der Zusammenstellung beteiligter Aktoren widmet sich dieser Abschnitt nunmehr den Anwendungsfällen. Die große Komplexität der Fragestellung impliziert freilich eine große Bandbreite derer. Im Sinne eines strukturierten Vorgehens wurde eine Aufteilung in mehrere Dimensionen vorgenommen, die ein möglichst weitgreifendes und vollständiges Bild der maßgeblichen Anforderungen an gridbasierte Intrusion Detection Systeme ergeben.

Abschnitt 4.3.1 befasst sich initial mit Anwendungsfällen, denen im Umfeld von Grids bedingt durch die neuartige, organisationsübergreifende Zusammenarbeit jederzeit Beachtung geschenkt werden muss. Daran knüpfen sich in Abschnitt 4.3.2 die gerade innerhalb des Szenarios spezifisch für eine Einbruchserkennung relevanten Anwendungsfälle an. Nicht außer Acht gelassen werden kann der Aspekt einer Integration in übrige Systeme, mit der sich sogleich Abschnitt 4.3.3 auseinandersetzt. Schließlich zeigt Abschnitt 4.3.4 überdies hinzukommende Anwendungsfälle, die im Sinne einer vollständigen Abdeckung grundsätzlich von Interesse sind.

Zur Abrundung stellt Abschnitt 4.3.5 noch einmal übersichtlich die Anwendungsfälle und die zugehörigen Anforderungen tabellarisch zusammen.

4.3.1 Organisationsübergreifende Zusammenarbeit

1. **Autonomie beteiligter Organisationen**

   Die Zusammenarbeit im Grid schließt ausdrücklich eine weit gefasste Autonomie beteiligter Unternehmen mitein, die sowohl technische, organisatorische und rechtliche Aspekte betrifft. Unter Beachtung dieses spezifischen Umfeldes soll gleichwohl eine übergreifende Einbruchserkennung gewährleistet werden.

   Resultierende Anforderungen:

   - Aggregation von Daten

- Anwendung site-spezifischer Richtlinien und Kollaborationsverträge (ggf. durch Anonymisierung)
- Nutzung standardisierter Datenformate
- Unterstützung heterogener Ressourcen
- Einheitliche Schnittstellen
- Portabilität
- Wiederverwendbarkeit
- Interoperabilität

2. **Datenschutz/Vertraulichkeit**

   Die virtuelle Organisation A führt aufwendige medizinische Modellberechnungen durch, wobei zeitweilig auch aktuelle personenbezogene Daten von Patienten miteinbezogen werden. Hierbei werden unter anderem Wirkstoffe zukünftiger, potenziell gewinnträchtiger Medikamente evaluiert. Im Unterschied dazu setzt die virtuelle Organisation D die leistungsfähigen Ressourcen des Grids für ausgedehnte Simulationen bezüglich neuer, noch streng geheimer Produkte eines führenden Technologieanbieters ein.
   Das anvisierte System soll die Vertraulichkeit von Informationen wahren und entsprechende Datenschutzvereinbarungen jederzeit einhalten.

   Resultierende Anforderungen:

   - Beachtung des Datenschutzes
   - Anwendung site-spezifischer Richtlinien und Kollaborationsverträge (ggf. durch Anonymisierung)
   - Einbeziehung übergreifender AA-Mechanismen
   - Gesicherter Informationsaustausch (Kryptografie/Signaturen)
   - Integrität
   - Aggregation von Daten
   - Nachvollziehbarkeit

3. **Vielzahl domänenübergreifender, heterogener Ressourcen**

   Ressourcenanbieter A stellt für die Nutzung im Grid freie Kapazitäten eines Großrechners mit proprietärem, herstellerspezifischem Betriebssystem bereit. Demgegenüber erlaubt Ressourcenanbieter B den Zugriff auf zwei getrennte Linux-Cluster (32-/64-Bit) unter verschiedenen Systemen mit jeweils mehr als 200 Knoten.
   Trotz der Mächtigkeit und Vielfalt des anzunehmenden Einsatzszenarios muss die Einbruchserkennung jederzeit nachhaltig sichergestellt sein.

   Resultierende Anforderungen:

   - Mehrdomänenfähigkeit
   - Unterstützung heterogener Ressourcen
   - Portabilität
   - Wiederverwendbarkeit
   - Interoperabilität
   - Integrität
   - Lastverteilung
   - Effizienz
   - Performanz

- Skalierbarkeit

4. **Virtuelle Ressourcen**

   Die virtuelle Organisation A umfasst unter anderem eine virtuelle Ressource, die mehrere homogene Cluster in sich vereint. Zusätzlich werden durch die virtuelle Organisation B mehrere weit verteilte Archivsysteme zu einer virtuellen Ressource gebündelt.
   Unabhängig davon soll die Einbruchserkennung zusätzlich zu realen auch virtuelle Ressourcen einbeziehen.

   Resultierende Anforderungen:

   - Unterstützung virtueller Organisationen
   - Ressourcen-Discovery
   - Skalierbarkeit
   - Interoperabilität
   - Portabilität
   - Integration virtueller Ressourcen

5. **Dynamik der Ressourcen**

   Ressourcenanbieter B verringert immer wieder die Anzahl der Knoten eines der ins Grid eingebrachten Linux-Cluster, falls sie durch das Grid nicht ausgelastet werden und stattdessen für ein anderweitiges Projekt hinzugezogen werden sollen. Gegebenenfalls entfällt der Cluster für das Grid vorübergehend auch vollständig. Ressourcenanbieter C kommt erstmalig hinzu, wobei dieser dem Grid einen weiteren Linux-Cluster hinzufügt.
   Hiervon unberührt muss das Intrusion Detection System jederzeit funktionsfähig bleiben.

   Resultierende Anforderungen:

   - Unterstützung einer hohen Dynamik einzubeziehender Ressourcen
   - Unterstützung virtueller Organisationen
   - Ressourcen-Discovery
   - Robustheit
   - Anpassungsfähigkeit
   - Erweiterbarkeit
   - Flexibilität

6. **Dynamik der Nutzer**

   Nach Abschluss eines umfangreichen Projekts verlässt die betreffende Nutzergruppe die virtuelle Organisation A. Dagegen erfordert ein weiteres im Aufbau befindliches Projekt das ständige Hinzukommen weiterer Mitglieder zur virtuellen Organisation B, wobei sich auch deren jeweilige Aufgaben noch verändern.
   Trotz der hohen Dynamik der Nutzer soll die Einbruchserkennung fortwährend relevante Angriffe nachvollziehen können.

   Resultierende Anforderungen:

   - Unterstützung einer hohen Dynamik berechtigter Nutzer
   - Unterstützung virtueller Organisationen
   - Nutzer-Discovery

4.3.2 Intrusion Detection in Grids

7. **Domänenübergreifende Einbruchserkennung**

   Ressourcenanbieter F überlässt dem Grid Datenbanken und Archivsysteme aus drei seiner verschiedenen lokalen und administrativ vollständig getrennten Domänen. Ressourcenanbieter K stellt ausschließlich Rechenressourcen aus zwei verschiedenen Domänen bereit.
   Als wichtige Zielsetzung muss eine domänen- und sogar organisationsübergreifende Einbruchserkennung gelten.

   Resultierende Anforderungen:

   - Mehrdomänenfähigkeit
   - Skalierbarkeit
   - Portabilität
   - Wiederverwendbarkeit
   - Interoperabilität
   - Einheitliche Schnittstellen
   - Gesicherter Informationsaustausch (Kryptografie/Signaturen)
   - Integrität
   - Unterstützung heterogener Ressourcen

8. **Existierende Ansätze zur Einbruchserkennung**

   Ressourcenanbieter A verlässt sich bislang auf die beiden herkömmlichen Intrusion Detection Systeme „Snort“ und „Samhain“, während Ressourcenanbieter B aktuell ausschließlich „Prelude“ heranzieht.
   Die anzustrebende Lösung soll unter Beachtung existierender Konzepte zur Einbruchserkennung übergreifend funktionieren.

   Resultierende Anforderungen:

   - Geringe Modifikation existierender Umgebungen
   - Einbindung bestehender Ansätze (durch Adaption bzw. Erweiterung)
   - Unterstützung etablierter Standards
   - Einheitliche, generische Schnittstellen
   - Standardisierte Datenaustauschformate

9. **Neuartige, durch das Grid induzierte Komponenten**

   Innerhalb der virtuellen Organisation A kommen derzeit mit dem „Globus Toolkit“ und „UNICORE“ aus Kompatibilitätsgründen gleich zwei verschiedene Middlewares zum Einsatz, um den Zugriff auf die mannigfachen Ressourcen für unter anderem Rechen- und Speicherzwecke organisationsübergreifend zu koordinieren. Dagegen berücksichtigt die virtuelle Organisation B als Middleware nur „gLite“. In beiden Fällen bestehen weitere gridbedingte Komponenten, beispielsweise für ein gridweites Monitoring sämtlicher Ressourcen und Dienste oder das Accounting.
   Die Einbruchserkennung muss bei Berücksichtigung neuartiger, durch das Grid induzierter Komponenten übergreifend sichergestellt werden.

   Resultierende Anforderungen:

   - Unterstützung etablierter Standards

- Einheitliche Schnittstellen
- Portabilität
- Wiederverwendbarkeit
- Interoperabilität
- Integration durch das Grid induzierter Komponenten
- Unterstützung heterogener Ressourcen

10. **Gridspezifische Bedrohungen**

    Einzelne Systeme des Ressourcenanbieters B waren mehrfach von erfolgreich durchgeführten Angriffen betroffen, wobei noch unbekannte Schwachstellen der jeweils eingesetzten Middleware ausgenutzt werden konnten. Innerhalb der virtuellen Organisation A kam es zu einem missbräuchlichen Einsatz von Ressourcen, wobei Fehler eines Werkzeugs zum Management virtueller Organisationen ursächlich waren.
    Die anzustrebende Einbruchserkennung soll komplexe Grids möglichst erschöpfend auf alle relevanten Angriffe hin überwachen.

    Resultierende Anforderungen:

    - Abdeckung gridspezifischer Angriffstypen
    - Minimale Fehlerrate
    - Modifizierbarkeit bzgl. neuer Bedrohungen
    - Integrität
    - Verifizierbarkeit

11. **Nutzergruppenspezifische Berichterstattung**

    In Gridumgebungen finden sich prinzipiell verschiedenartige Nutzergruppen, denen im Rahmen der Entwicklung eines gridspezifischen Intrusion Detection Systems umfassend Rechnung getragen werden muss.
    Neben einem zentralen „Grid Operations Center“ und den Ressourcenanbietern sollen auch virtuelle Organisationen (gegebenenfalls einzelne autorisierte Mitglieder) Zugriff auf vom System generierte Berichte erhalten, um so jederzeit die Sicherheitslage beurteilen zu können. Einzelne Nutzergruppen sollen dabei möglicherweise nur individuell eingeschränkte, vorher vereinbarte Informationen beziehen können, wobei maßgebliche Datenschutz- und Sicherheitsrichtlinien beachtet werden müssen.

    Resultierende Anforderungen:

    - Unterstützung virtueller Organisationen
    - Einbeziehung übergreifender AA-Mechanismen
    - Anwendung site-spezifischer Richtlinien und Kollaborationsverträge
    - Nutzung standardisierter Datenformate
    - Verschiedene Granularitätsstufen
    - Mandantenfähigkeit, nutzergruppenabhängige Sichten
    - Einheitliche Bedienoberfläche (z.B. GUI)
    - Konfigurierbarkeit
    - Modifizierbarkeit
    - Nachvollziehbarkeit durchgeführter Anfragen
    - Transparenz für den Anwender
    - Aussagekräftige Informationsaufbereitung
    - Nachhalten historischer Daten

4.3.3 Integration in übrige Systeme

12. **Management des Systems**

    Zur Unterstützung komplexer Managementprozesse kann das „Grid Operations Center“ auf eine Sammlung anerkannter Werkzeuge zurückgreifen. Die einzelnen Ressourcenanbieter verwenden ebenfalls individuelle Managementwerkzeuge.
    Wichtige Managementaufgaben einer gridbasierten Einbruchserkennung, beispielsweise betreffend die Administration oder die Auswertung von aufgelaufenen Informationen, sollen im Bedarfsfall leicht in vorhandene Lösungen im Kontext des Grids einzugliedern sein.

    Resultierende Anforderungen:

    - Integrierbarkeit in bestehende Managementwerkzeuge
    - Interoperabilität
    - Unterstützung etablierter Standards
    - Einheitliche Schnittstellen
    - Gesicherter Informationsaustausch (Kryptografie/Signaturen)
    - Einheitliche Datenaustauschformate

13. **Monitoring des Systems**

    Ressourcenanbieter E, von dem das Grid eine Reihe von Ressourcen bezieht, verfügt bereits über ein hinreichend ausgereiftes, auf die anspruchsvollen Bedürfnisse des Grids zugeschnittenes Monitoringsystem. Die aufwendige, weitgreifende Eigenentwicklung fasst Informationen aus mehreren verteilten Quellen zusammen, wobei auch verbreitete Monitoring-Frameworks genutzter Middlewares eingebunden werden.
    Ein gegebenenfalls schon vorhandenes Monitoringsystem soll den gegenwärtigen Status des gridbasierten Intrusion Detection Systems fortwährend beobachten können.

    Resultierende Anforderungen:

    - Überwachbarkeit
    - Interoperabilität
    - Unterstützung etablierter Standards
    - Einheitliche Schnittstellen
    - Gesicherter Informationsaustausch (Kryptografie/Signaturen)
    - Einheitliche Datenaustauschformate

4.3.4 Ergänzende, grundlegende Anwendungsfälle

14. **Analyseverfahren**

    Um über potenzielle Angriffe frühzeitig in Kenntnis gesetzt werden zu können, soll die Auswertung sicherheitsrelevanter Ereignisse permanent, möglichst „nahe Echtzeit“ erfolgen. Daneben soll aber auch eine bedarfsorientierte Untersuchung aufgelaufener Daten im Nachhinein ermöglicht werden, um so auch neue, bislang nicht erkannte Ereignisse und Trends erkennen zu können.

    Resultierende Anforderungen:

    - Aggregation der Daten
    - Korrelation der Daten

- Minimale Fehlerrate
- Effizienz
- Performanz
- „Echtzeitfähigkeit“
- Unterstützung einer bedarfsgesteuerten Analyse im Nachhinein
- Nachhalten historischer Daten

15. **Datenzugriffsmodell**

    Ein IDS-Administrator soll aktiv die angefallenen Daten zu sicherheitskritischen Ereignissen einsehen und untersuchen können. Daneben möchte er aber auch bei wichtigen Vorkommnissen direkt benachrichtigt werden, um gegebenenfalls kurzfristig reagieren zu können. Es sollen also sowohl ein bedarfsweises Abholen von Informationen, als auch eine proaktive Alarmierung unterstützt werden.

    Resultierende Anforderungen:

    - Einheitliche Bedienoberfläche (z.B. GUI)
    - Mandantenfähigkeit
    - Einbeziehung übergreifender AA-Mechanismen
    - Pull-/Push-Datenzugriffsmodell

16. **Administration**

    Intrusion Detection Systeme erfordern aufgrund ihrer Komplexität für einen dauerhaft zuverlässigen Betrieb die fortwährende Nachführung (*Kalibrierung*) und Überwachung relevanter Parameter, etwa bezüglich der Sensitivität zu erfassender Meldungen oder zur Protokollierung oder Alarmierung.
    Zuständige Administratoren sollen daher jederzeit, auch zur Laufzeit des Systems und insbesondere zur Berücksichtigung von Veränderungen der Einsatzumgebung Anpassungen vornehmen können.

    Resultierende Anforderungen:

    - Konfigurierbarkeit
    - Kalibrierbarkeit
    - Einheitliche Bedienoberfläche (z.B. GUI)
    - Nachvollziehbarkeit durchgeführter Änderungen
    - Transparenz für den Anwender

17. **Benutzerschnittstelle**

    Die unterschiedlichen potenziellen Anwender des angestrebten gridweiten Intrusion Detection Systems erwarten, dass sämtliche Werkzeuge unkompliziert anzuwenden sind.

    Resultierende Anforderungen:

    - Einheitliche Bedienoberfläche (z.B. GUI)
    - Einbeziehung übergreifender AA-Mechanismen
    - Nutzung standardisierter Datenformate
    - Transparenz für den Anwender
    - Aussagekräftige Informationsaufbereitung

18. **Reduktion manueller Eingriffe**

    Um den zuverlässigen Betrieb des Systems zu gewährleisten, soll die Einbruchserkennung weitgehend ohne die Erfordernis manueller Eingriffe erfolgen.
    Administratoren fordern daher, dass die für die Erhaltung der Funktionalität notwendigen Interaktionen auf ein Minimum reduziert bleiben.

    Resultierende Anforderungen:

    - Autonomie der Einbruchserkennung
    - Minimale Fehlerrate
    - Zuverlässigkeit
    - Robustheit/Unverletzlichkeit

19. **Erweiterbarkeit/Flexibilität**

    Auch nach der Inbetriebnahme der Einbruchserkennung kann zur Laufzeit ein Ausbau des Systems erforderlich werden. Ressourcenanbieter A möchte der Architektur weitere Komponenten (Sensoren) hinzufügen, um die Leistungsfähigkeit zu verbessern. Die virtuelle Organisation A möchte zudem weitere, projektbezogene Berichte erhalten. Daneben ändern sich die Kollaborationsverträge der Ressourcenanbieter A und D.
    Das Intrusion Detection System soll hierfür jederzeit ohne großen Aufwand angepasst und erweitert werden können.

    Resultierende Anforderungen:

    - Skalierbarkeit
    - Wartbarkeit/Dokumentation
    - Erweiterbarkeit
    - Modifizierbarkeit
    - Konfigurierbarkeit
    - Flexibilität

20. **Dauerhafte Stabilität/Verfügbarkeit des Systems**

    Die virtuelle Organisation A, die eine große Anzahl an Ressourcen im Grid anbietet, fordert, dass trotz des Ausfalls oder einer unerwarteten Fehlfunktion einer Komponente des Intrusion Detection Systems weiterhin eine dauerhafte Stabilität und hohe Verfügbarkeit gewährleistet bleiben.

    Resultierende Anforderungen:

    - Robustheit/Unverletzlichkeit
    - Lastverteilung

21. **Beweissicherung**

    Ressourcenanbieter A verzeichnete Einbrüche in wichtige Systeme des Grids, wobei sogar wiederholt versucht wurde, die durchgeführten Aktionen im Nachhinein abzustreiten.
    Von einer gridweiten Lösung soll erwartet werden, dass sie sämtliche relevanten Beweise zu durchgeführten Angriffen möglichst vollumfänglich sichert.

    Resultierende Anforderungen:

- Nachhalten historischer Daten
- Integrität
- Verifizierbarkeit

4.3.5 Übersicht über die Anforderungen

Zur Beurteilung der Relevanz einzelner Anforderungen an gridbasierte Intrusion Detection Systeme erscheint es zweckmäßig, jeweils zu untersuchen, in welchen und besonders auch in wie vielen der Anwendungsfälle sie aufgeführt sind. Tabelle 4.1 unterstützt dies, indem die einzelnen Anforderungen und Anwendungsfälle übersichtlich zusammengestellt werden. Die häufigsten Anforderungen finden sich dabei zuerst.

4.4 Abgeleitete generische Anforderungen

Im vorausgegangenen Abschnitt 4.3 wurden bereits aus maßgeblichen Anwendungsfällen grundlegende Anforderungen abgeleitet. Nachfolgend werden diese nun jeweils separat erklärt, sowie mit einer Erläuterung ihrer konkreten Relevanz versehen. Die Anordnung richtet sich dabei nach Tabelle 4.1.

Interoperabilität

Neben der Portabilität bedeutet auch die Interoperabilität innerhalb hochdynamischer, hochgradig heterogener Grid-Umgebungen eine entscheidende Anforderung an die zu entwickelnde Lösung. Innerhalb eines gridweiten Intrusion Detection Systems müssen schon aufgrund der oftmals weit gefassten Autonomie beteiligter Organisationen äußerst verschiedenartige Komponenten Berücksichtigung finden. Insbesondere betrifft dies die potenzielle Kommunikation unterschiedlicher Komponenten des Systems untereinander, wobei deren reibungslose Kooperation wiederum durch einheitliche Schnittstellen und durch geeignete, standardisierte Datenaustauschformate gefördert werden kann.

Einheitliche Schnittstellen

Die übergreifende Verfügbarkeit einheitlicher Schnittstellen erleichtert den Informationsaustausch größtmöglich, insbesondere unter dem für das Grid entscheidenden Teilaspekt der großen Heterogenität auf vielen Ebenen. Ein integratives Intrusion Detection System muss auf dem Transfer von Informationen zwischen unterschiedlichen Komponenten basieren. Ebenso sollen auch externe Dienste zu unterschiedlichen Zwecken (Monitoring etc.) Zugriff auf vom System generierte Daten erhalten. Die Anwendung einheitlicher Schnittstellen bezieht sich dabei neben Programmierschnittstellen (engl. *Application Programming Interface (API)*) auch auf standardisierte, möglichst offen gelegte Protokolle.

Integrität

Bezüglich der Behandlung sicherheitskritischer Informationen muss vor allem auch die fortwährende Wahrung der Integrität garantiert bleiben. Einerseits soll eine unautorisierte Datenmodifikation möglichst verhindert oder andernfalls wenigstens sicher als solche aufgedeckt werden (zum Beispiel

Anforderung	Anwendungsfall																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Interoperabilität	✓		✓	✓			✓		✓			✓	✓								
Einheitliche Schnittstellen	✓						✓	✓	✓			✓	✓								
Integrität		✓	✓				✓			✓											✓
Nachvollziehbarkeit/Verifizierbarkeit		✓								✓	✓					✓					✓
Portabilität und Wiederverwendbarkeit	✓		✓	✓			✓		✓												
Einbeziehung übergr. AA-Mechanismen		✓									✓				✓		✓				
Einheitliche Bedienoberfläche											✓				✓	✓	✓				
Gesicherter Informationsaustausch		✓					✓					✓	✓								
Konfigurierbarkeit/Modifizierbarkeit										✓	✓					✓			✓		
Skalierbarkeit			✓	✓			✓												✓		
Unterstützung etablierter Standards								✓	✓			✓	✓								
Unterstützung heterogener Ressourcen	✓		✓				✓		✓												
Unterstützung virtueller Organisationen				✓	✓	✓					✓										
Aggregation der Daten	✓	✓												✓							
Anwendung spez. Richtlinien u. Verträge	✓	✓									✓										
Minimale Fehlerrate										✓				✓				✓			
Nachhalten historischer Daten											✓			✓							✓
Nutzung einheitlicher Datenformate	✓										✓						✓				
Robustheit/Unverletzlichkt./Zuverlässigkt.					✓													✓		✓	
Standardisierte Datenaustauschformate								✓				✓	✓								
Transparenz für den Anwender											✓					✓	✓				
Architektur/Lastverteilung			✓																	✓	
Aussagekräftige Informationsaufbereitung											✓						✓				
Effizienz/Performanz			✓											✓							
Erweiterbarkeit/Flexibilität					✓														✓		
Mandantenfähigkeit											✓				✓						
Mehrdomänenfähigkeit			✓				✓														
Ressourcen-Discovery				✓	✓																
Abdeckung gridspezifischer Angriffstypen										✓											
Anpassungsfähigkeit					✓																
Autonomie der Einbruchserkennung																		✓			
Beachtung des Datenschutzes		✓																			
„Echtzeitfähigkeit“														✓							
Einbindung bestehender Ansätze								✓													
Geringe Modifikation exist. Umgebungen								✓													
Integration Grid-induzierter Komponenten									✓												
Integration virtueller Ressourcen				✓																	
Integrierbarkeit in Managementwerkzge.												✓									
Kalibrierbarkeit																✓					
Korrelation der Daten														✓							
Nutzer-Discovery						✓															
Pull-/Push-Datenzugriffsmodell															✓						
Überwachbarkeit													✓								
Unterstützg. der Analyse im Nachhinein														✓							
Unterstützung hoher Nutzer-Dynamik						✓															
Unterstützg. hoher Ressourcen-Dynamik					✓																
Verschiedene Granularitätsstufen											✓										
Wartbarkeit/Dokumentation																			✓		

Tabelle 4.1: Übersicht über die Anforderungen und Anwendungsfälle

durch die Nutzung kryptografischer Prüfsummen). Für letzteren Fall müssen gegebenenfalls geeignete Recovery-Maßnahmen (Backups etc.) vorgesehen werden, um jederzeit wieder in einen integeren Zustand gelangen zu können.

Nachvollziehbarkeit/Verifizierbarkeit

Als zusätzliche Anforderung an das gesamte System ist eine tief greifende Nachvollziehbarkeit beziehungsweise Verifizierbarkeit zu stellen, wodurch gleichzeitig die separate Anforderung des Nachhaltens historischer Daten einhergeht. Auf diese Weise sollen die Funktionsfähigkeit der Angriffserkennung sowie die von Nutzern durchgeführten Anfragen jederzeit verstanden und überprüft werden können.

Portabilität und Wiederverwendbarkeit

Innerhalb anzunehmender, hochdynamischer Grid-Szenarien muss generell mit einem hohen Maß an Heterogenität auf vielen Schichten, wie beispielsweise betreffend die Ressourcen, Middlewares et cetera gerechnet werden. An vorderster Stelle muss eine Einbruchserkennung im Grid daher portable, das heißt hochgradig plattformunabhängige Komponenten integrieren, um den Aufwand einer potenziell erforderlichen Portierung möglichst gering zu halten und die Wiederverwendbarkeit zu garantieren. Dies betrifft vornehmlich die über das Grid verteilten Sensoren, daneben aber auch die übrigen Bausteine eines übergreifenden Intrusion Detection Systems. Die Portabilität kann gegebenenfalls unterstützt werden, indem diszipliniert vergleichbare Schnittstellen zur Anwendung kommen und einheitliche, standardisierte Formate für die Speicherung und den Austausch sicherheitsrelevanter Daten genutzt werden.

Einbeziehung übergreifender AA-Mechanismen

Eine spezielle Infrastruktur bezüglich der Authentifikation und Autorisierung (AAI) stellt einen notwendigen und komplexen Bestandteil jeder verteilten Grid-Umgebung dar. Insofern ist es auf jeden Fall geboten, dass auch ein Frühwarnsystem in einer diesbezüglichen Umgebung daran weitestgehend angepasst ist, wobei sich derartige Mechanismen in verschiedenen Szenarien jedoch durchaus unterscheiden.

Einheitliche Bedienoberfläche

Zur Gewährleistung eines hohen Maßes an Benutzbarkeit der zu schaffenden Lösung ist eine einheitliche, einfach zu bedienende Oberfläche etwa in Form einer grafischen Benutzerschnittstelle (engl. *Graphical User Interface (GUI)*) für die Interaktion mit dem System von essenzieller Bedeutung. Es müssen sowohl die Konfiguration des Systems als auch die Auswertung von Meldungen bedacht werden. Zur Untersuchung von IDS-Meldungen kann es sinnvoll sein, extern erreichbare Quellen einzubetten, um Hintergrundinformationen zu Angriffen zu erhalten oder gegebenenfalls auch eigene Beobachtungen dorthin zu melden. Gerade im Kontext übergreifender Grids kann auch die Integration der Bedienoberfläche in etwaige bereits bestehende, übergeordnete Managementwerkzeuge sinnvoll erscheinen.

Gesicherter Informationsaustausch (Kryptografie/Signaturen)

Mithilfe kryptografischer Verfahren sollten Informationen soweit relevant verschlüsselt werden, um deren Vertraulichkeit sicherstellen zu können. Digitale Signaturen können zudem die Urheberschaft jederzeit überprüfbar nachweisen.

Konfigurierbarkeit/Modifizierbarkeit

Die anzustrebende Lösung sollte auf Basis einer unterstützenden Benutzerschnittstelle abgesehen von einer Kalibrierung weiterreichende Möglichkeiten der Konfiguration anbieten, um das Systemverhalten initial und zur Laufzeit auf einfache, dokumentierte Weise umfassend modifizieren und beeinflussen zu können.

Skalierbarkeit

Neben der im Umfeld von Grids auftretenden potenziell sehr großen Zahl an domänenübergreifenden Ressourcen und Diensten ist zumeist eine gleichfalls potenziell umfangreiche Menge an Nutzern involviert. Die Einbruchserkennung muss daher sowohl hinsichtlich der zu integrierenden Ressourcen als auch der einzubeziehenden Nutzer skalierbar sein.

Unterstützung etablierter Standards

Eine wichtige Anforderung an das zu schaffende System besteht grundsätzlich noch vor jedweder Entwicklung eigener, proprietärer Konzepte in der Unterstützung bereits anerkannter und etablierter Standards, wovon in jüngster Zeit viele entwickelt und hervorgebracht wurden. Insbesondere erscheint es sinnvoll, architekturspezifische Standards (OGSA, OGSI, WSRF etc.) zu beachten, um die Wiederverwendbarkeit und Interoperabilität des Ansatzes nachhaltig sicherstellen zu können.

Unterstützung heterogener Ressourcen

Im Grid sind folgend aus der weit gefassten Autonomie beteiligter Organisationen verglichen mit typischen lokalen Umgebungen oftmals Ressourcen verknüpft, die womöglich überaus heterogen sind. Komponenten können sich stark in Hardware, Software und Netzanbindung unterscheiden. Das aufzuzeigende System muss daher die Einbruchserkennung in einer potenziell äußerst heterogenen Landschaft erbringen.

Unterstützung virtueller Organisationen

Grundsätzlich stellt die transparente Nutzung von Ressourcen und Diensten über viele, auch geografisch verteilte und administrativ unabhängige Organisationen eine große Herausforderung bei der Betrachtung von Grids dar. Die Berücksichtigung und Unterstützung des maßgeblichen Konzepts virtueller Organisationen ist daher gleichfalls zentral für eine integrative, gridübergreifende Einbruchserkennung, wobei dies insbesondere die umfassende Einbeziehung vielfältiger etablierter Systeme bezüglich des Managements virtueller Organisationen erfordert.

Aggregation der Daten

Im Rahmen des Betriebs einer gridübergreifenden Einbruchserkennung führt schon allein die Mächtigkeit des anzunehmenden Einsatzszenarios zu einer potenziell eher umfangreichen Menge an sicherheitsrelevanten Informationen. Insbesondere auf Sensorebene muss mit einem anhaltend großen Datenaufkommen gerechnet werden. Um unter diesen Umständen eine hohe Leistungsfähigkeit des Systems gewährleisten zu können, sollten die anfallenden Daten bereits frühzeitig sinnvoll aggregiert werden, wobei wichtige Details jedoch nicht verloren gehen dürfen. Einerseits kann dies zwar zu einem geringfügig gestiegenen Aufwand auf niedrigen Stufen führen. Andererseits lässt sich dadurch aber das auftretende Datenaufkommen immens reduzieren, wovon letztlich auch direkt die weitere Verarbeitung und Analyse der Informationen profitiert.

Anwendung site-spezifischer Richtlinien und Kollaborationsverträge

Die beteiligten Organisationen verfügen in der Regel bereits über diverse lokale Richtlinien (Sicherheitspolitik etc.), deren Anwendung hinsichtlich der Konzeption von Intrusion Detection Systemen zwingend berücksichtigt werden muss. Des weiteren müssen Verträge angemessen berücksichtigt werden, die die Zusammenarbeit der unterschiedlichen Partner im Grid regeln.

Minimale Fehlerrate

Eine minimale Fehlerrate stellt eine essenzielle Anforderung an jedwedes Intrusion Detection System im Allgemeinen dar. Eine hohe Fehlerrate führt zwangsläufig zu einem hohen Aufwand in der „Nachbearbeitung“ aufgelaufener Meldungen, wodurch auch der Aspekt der Autonomie der Einbruchserkennung infrage gestellt wird. Werden Angriffe gar nicht erst erkannt, kann dabei ein realer, nicht zu beziffernder Schaden eintreten. Die Güte der Kriterien zur Bestimmung relevanter Angriffe bestimmt damit direkt den Erfolg und den Nutzen von Intrusion Detection Systemen.

Nachhalten historischer Daten

Das Nachhalten historischer Ereignisdaten, sowie damit etwaig verbundener Kontextinformationen, ist als integrale Aufgabe einer anzustrebenden integrativen Gesamtlösung zu erachten. Der Umfang anzunehmender Grid-Szenarien bedingt gleichzeitig jedoch eine potenziell äußerst große Menge an zu verarbeitenden Daten, wodurch womöglich nur eine vorübergehende und/oder reduzierte Erfassung sinnvoll oder umsetzbar erscheinen kann. Vorwiegend muss die Speicherung dem Zwecke der Nachvollziehbarkeit und Verifizierbarkeit aller relevanten Vorgänge dienen. Ebenso ist sie für die Unterstützung einer bedarfsgesteuerten Analyse von Angriffen im Nachhinein unverzichtbar. Insbesondere unterstützt sie aber auch forensische Anstrengungen, um die Spuren krimineller Angreifer zumindest für einen wohl definierten Zeitraum zu sichern. Nicht außer Acht gelassen werden können Überlegungen betreffend den Datenschutz. Unter Umständen muss auch der Zugriff auf historische Daten durch externe Anwendungen (Monitoring etc.) in Betracht gezogen werden.

Nutzung einheitlicher Datenformate

Die Verwendung etablierter und einheitlicher Datenformate trägt mit dazu bei, organisationsübergreifend ein Höchstmaß an Interoperabilität zu gewährleisten und außerdem die Nachhaltigkeit des Systems sicherzustellen.

Robustheit/Unverletzlichkeit/Zuverlässigkeit

Die Anforderung nach einem Höchstmaß an Ausfallsicherheit impliziert gleichzeitig die Robustheit der anzustrebenden Lösung. Das gridbasierte Intrusion Detection System soll tolerant gegenüber dem Ausfall oder einer unerwarteten Fehlfunktion einzelner Komponenten sein, um somit dauerhaft eine hohe Verfügbarkeit gewährleisten zu können. Dies erfordert einerseits den Einsatz eines gezielten, planvollen Fehlermanagements sowie vor allem auch geeigneter Mechanismen für die zeitnahe Wiederherstellung, beispielsweise durch Replikation aller relevanten Komponenten.

Standardisierte Datenaustauschformate

Neben einheitlichen Formaten für die Speicherung von Daten spielen auch standardisierte Datenaustauschformate für Intrusion Detection Systeme eine wichtige Rolle. Für eine reibungslose Kommunikation sind einheitliche und offene Lösungen unabdingbar.

Transparenz für den Anwender

Ein praktikables Höchstmaß an Transparenz für den Anwender stellt natürlich eine grundsätzliche Anforderung an jedwedes Softwaresystem dar. Gerade innerhalb von Grids erlangt dieser Aspekt aber eine besondere Bedeutung. Die aus Sicht des Anwenders übersteigerte Informationsfülle eines derart vielschichtigen Produkts, wie das einer gridübergreifenden Einbruchserkennung, sollte auf das jeweils erforderliche Mindestmaß reduziert bleiben, um die Interaktion mit dem System nicht unnötig zu erschweren. Im vorliegenden Fall betrifft dies beispielsweise die Vermeidung zu vieler ereignisspezifischer Einzelheiten an der Benutzerschnittstelle.

Architektur/Lastverteilung

Gerade für sicherheitskritische Systeme, wie etwa solche zur Einbruchserkennung, ist deren ständige Verfügbarkeit eine unumgängliche und zentrale Anforderung. Auch kurzfristige Ausfälle können zu nicht kalkulierbaren und mitunter auch wirtschaftlich immensen Folgen führen. Dies stellt natürlich auch besondere Anforderungen an den Entwurf betreffend die Architektur. Insbesondere kann die Verfügbarkeit durch den Einsatz ausgefeilter Mechanismen der Lastverteilung verbessert werden, wie sie auch von verbreiteten Middlewares bereitgestellt werden. Hierbei teilen sich mehrere Systeme die Aufgabenerbringung, wobei ausgefallene Systeme gleichfalls automatisch erkannt und umgangen werden können. Natürlich dürfen zentrale Systeme zur Verteilung der Last an untergeordnete Systeme aber keinen Flaschenhals darstellen.

Aussagekräftige Informationsaufbereitung

Eine möglichst aussagekräftige Aufbereitung von Informationen (unter anderem zu festgestellten Ereignissen) verbessert klar ersichtlich die Möglichkeiten zur Einordnung und Bewertung relevanter Vorfälle. Hierzu kann im gegebenen Fall ein Rückgriff auf diejenigen der Meldung zugrunde liegenden Daten (etwa von den Sensoren) erforderlich werden, was dann deren nachhaltige Verfügbarkeit verlangt. Unter Umständen kann zusätzlich ebenso die Einbindung externer Quellen von Nutzen sein. Der Umfang und Detailgrad bereitgestellter Informationen sollten sich hierbei nach dem spezifischen Einsatzszenario und der Art der betreffenden Ereignisse bemessen.

Effizienz/Performanz

Fragen der Effizienz und Performanz betreffen hauptsächlich die Auswertung aufgelaufener Ereignisse innerhalb großflächiger Grid-Szenarien, wobei auch übrige Aufgaben des Intrusion Detection Systems betroffen sein können. Die Anforderungen korrespondieren mit der Anforderung zur „Echtzeitfähigkeit“.

Erweiterbarkeit/Flexibilität

Das zu entwickelnde Intrusion Detection System muss immer im Kontext des hochdynamischen Grid-Umfeldes gesehen werden. Damit Anpassungen an neuartige Gegebenheiten, wie beispielsweise größere Veränderungen in der Zusammensetzung und Verteilung der Ressourcen oder bezüglich der anzuwendenden Kollaborationsverträge, einfach umzusetzen sind, bedarf es auch vor dem Hintergrund der Nachhaltigkeit einer einfachen Erweiterbarkeit und größtmöglicher Flexibilität des Systems. Ein hoher Grad an Wartbarkeit/Dokumentation ist hierbei natürlich von Vorteil.

Mandantenfähigkeit

Die Unterstützung einer Vielzahl an möglichen, ebenso organisationsübergreifenden Nutzergruppen (Ressourcenanbieter, virtuelle Organisationen etc.) erfordert eine durchgängig implementierte Mandantenfähigkeit der anzustrebenden Gesamtlösung. An vorderster Stelle sind hierzu nutzergruppenabhängige Sichten auf die vom System bereitgestellten Informationen von Nutzen, wobei maßgebliche Sicherheitsrichtlinien im Hinblick auf die Auswahl bereitzustellender Daten anzuwenden sind.

Mehrdomänenfähigkeit

Im Mittelpunkt bei der Betrachtung des Grid Computing steht die für den Nutzer vollständig transparente Inanspruchnahme von Ressourcen und Diensten über administrative und organisatorische Grenzen hinweg. Dieser großen Herausforderung muss begegnet werden, indem eine gridbasierte Einbruchserkennung die Fähigkeit besitzt, mehrere Domänen, auch organisationsübergreifend und demnach nicht zwingend unter einheitlicher Kontrolle stehend, zu integrieren, um auf diese Weise ein kollaboratives Gesamtsystem für den übergreifenden Einsatz zu schaffen. Es muss diesbezüglich beachtet werden, dass hier ebenfalls von einer, wenn auch im Vergleich zu den einzelnen Ressourcen begrenzten Dynamik auszugehen ist und sich die Zusammensetzung einzubeziehender Domänen somit ändern kann.

Ressourcen-Discovery

Für der Erbringung einer wirkungsvollen Angriffserkennung ist die genaue Kenntnis der jeweils im Grid bereitgestellten Ressourcen oder Dienste sowie deren logischer Zusammenhang ein gewichtiger Teilaspekt. Gegebenenfalls muss hierfür eine Integration von relevanten Managementwerkzeugen (zum Beispiel etablierter Monitoringsysteme) angestoßen werden, die systematisch Informationen bezüglich des aktuellen Status von Ressourcen und Diensten sammeln, sowie diese Daten für nachträgliche Auswertungen auch nachhalten.

Abdeckung gridspezifischer Angriffstypen

Impliziert durch die neuartige Zusammenarbeit im Grid sieht man sich neben der Bedrohung mit herkömmlichen und hinreichend bekannten, genauso auch in weniger umfangreichen Umgebungen auftretenden Angriffstypen zunehmend mit einem neuen, gridspezifischen Angriffspotenzial konfrontiert, das von einfachen host- oder netzbasierten Intrusion Detection Systemen nicht oder nicht vollständig abgedeckt werden kann. Dies betrifft sowohl Angriffe von außen als auch von innen. Für eine wirksame Einbruchserkennung ist in jedem Fall eine fundierte, möglichst umfassende Modellbildung zu Grid-Vorfällen elementare Vorbedingung, aus der sich dann die relevanten Angriffstypen spezifisch für das jeweilige Szenario ableiten lassen.

Anpassungsfähigkeit

Über einen größtmöglichen Grad an Erweiterbarkeit und Flexibilität hinaus sollte das geplante Frühwarnsystem selbst über ein gewisses, von der Dynamik des Einsatzumfeldes abhängiges Maß an Anpassungsfähigkeit verfügen, um manuelle Eingriffe auf ein notwendiges Minimum zu reduzieren.

Autonomie der Einbruchserkennung

Ein nicht zu verachtender Aspekt bei der Untersuchung der Nutzbarkeit der Einbruchserkennung unter den besonderen Bedingungen groß angelegter Grids besteht in einer autarken Einbruchserkennung, die weitgehend ohne manuelle Eingriffe fortbesteht. Vor allem wird dies effektive Mechanismen für den Umgang mit erwarteten und ebenso unerwarteten Fehlersituationen erfordern. Ein nicht oder nur wenig selbstständig arbeitendes System kann besonders im Hinblick auf das Verhältnis zwischen Aufwand und Nutzen zur völligen Ablehnung des Verfahrens im Ganzen führen.

Beachtung des Datenschutzes

Die im Umfeld von Grids auftretenden Daten sind in vielen Fällen vertraulich oder im gegebenen Fall auch wertvoll und unterliegen dementsprechend regelmäßig hohen Anforderungen betreffend den Datenschutz. Dies schließt möglicherweise einen gesicherten Informationsaustausch in Form einer durchgängig verwendeten Datenverschlüsselung mitein. Eine gridübergreifende Erkennung von Angriffen muss daher zweifellos vertrauliche Daten vor unautorisiertem Zugriff schützen. Hierzu bedarf es der Einhaltung von für die einzelnen Institutionen spezifischen Richtlinien und von interorganisationellen Kollaborationsverträgen. Gerade die separate Anforderung bezüglich des Nachhaltens historischer Daten kann mit der Einhaltung des Datenschutzes kollidieren. Insbesondere ist zu klären, welche Daten für welchen Zeitraum gespeichert werden dürfen. Unter Umständen können Daten zu sicherheitsrelevanten Ereignissen auch in anonymisierter Form zur Erkennung von Einbrüchen beitragen.

„Echtzeitfähigkeit“

Beim Einsatz eines Intrusion Detection Systems entsteht ein wirklicher sicherheitstechnischer Nutzen zumeist erst dann, wenn es dadurch möglich wird, auf Ereignisse zeitnah und angemessen zu reagieren. Hierfür ist es unabdingbar, dass eine Auswertung sicherheitsrelevanter Ereignisse permanent und möglichst „nahe Echtzeit“ erfolgt.

Einbindung bestehender Ansätze (durch Adaption bzw. Erweiterung)

Sinnvolle, bestehende Ansätze sollen gegebenenfalls durch deren Anpassung oder Erweiterung flexibel in das anzustrebende gridweite Intrusion Detection System integriert werden können.

Geringe Modifikation existierender Umgebungen

In typischen Grid-Szenarien schließen sich viele Ressourcenanbieter zusammen. In deren bestehenden Umgebungen sind derzeit auch bereits häufig differierende Mechanismen für die Überwachung auf sicherheitsrelevante Ereignisse etabliert, denen jeweils individuelle Sicherheitsrichtlinien zugrunde liegen können. Trotz der bekannten gridbezogenen Defizite werden in Ermangelung gridbasierter Verfahren hierfür herkömmliche host- oder netzbasierte Einbruchserkennungssysteme eingesetzt. Um die Einführung neuartiger Lösungen wirksam zu unterstützen oder im Einzelfall sogar erst zu ermöglichen, besteht in jedem Fall eine wichtige Anforderung in einer geringen, zumutbaren Modifikation existierender Umgebungen.

Integration durch das Grid induzierter Komponenten

Um das Betriebsmodell des Grids auf herkömmliche Infrastrukturen zu projizieren, bedarf es deren Erweiterung mittels neuer durch das Grid induzierter Komponenten. Zuvorderst muss hier die inhärente Verwendung spezieller, möglicherweise auch heterogener Grid-Middlewares für den koordinierten Zugriff auf verteilte Ressourcen angeführt werden. Daneben kommen weitere Bausteine hinzu, etwa um unterschiedliche Managementaufgaben innerhalb des Grids zu übernehmen. Eine wirkungsvolle Erkennung von Angriffen bedeutet daher maßgeblich auch eine umfassende Einbeziehung sämtlicher durch das Grid erstmals hinzugekommener Komponenten.

Integration virtueller Ressourcen

Neben realen Ressourcen unterstützen die verbreiteten Grid-Umgebungen ebenso virtuelle Ressourcen. Solche virtuellen Ressourcen setzen sich zumeist aus zahlreichen realen Ressourcen (zum Beispiel mehreren Clustern) zusammen, auf die dann durch die Benutzer ohne Kenntnis der zugrunde liegenden Infrastruktur transparent zugegriffen werden kann. Virtuelle Ressourcen können dabei jederzeit erzeugt oder zerstört werden und sind gegebenenfalls zur Laufzeit auch veränderlich. Da gleichfalls Angriffe unter Ausnutzung des Konzepts virtueller Ressourcen denkbar sind, muss eine gridübergreifende Angriffserkennung hierauf eingehen.

Integrierbarkeit in bestehende Managementwerkzeuge

Zur Behandlung verbreiteter Managementaufgaben verfügen die beteiligten Organisationen zumeist schon über einen gewachsenen Fundus an etablierten Managementwerkzeugen. Eine vollständige oder teilweise Eingliederung wichtiger Managementaufgaben einer Einbruchserkennung in bestehende Werkzeuge kann hierbei die Einführung und den Betrieb vereinfachen und dauerhaft unterstützen. Generell setzt ein solches Vorgehen die Beachtung etablierter Standards voraus. Neben der Existenz einheitlicher Schnittstellen sind im gegebenen Fall auch standardisierte Datenformate von Bedeutung.

Kalibrierbarkeit

Gerade im Umfeld hoch dynamischer Grids muss der Kalibrierbarkeit große Bedeutung zugemessen werden. Im Rahmen der Kalibrierung wird kontinuierlich die Anpassung und Nachführung ausschlaggebender, erfolgbestimmender Parameter der Einbruchserkennung an die jeweils zutreffende Einsatzumgebung vorgenommen. Neben der Spezifikation relevanter Ereignisse bezüglich der eingebundenen Sensoren lässt sich hierbei auch festlegen, wie gegebenenfalls bei erkannten Ereignissen verfahren werden soll (Protokollierung, Alarmierung). Im Sinne einer vertretbaren, minimalen Fehlerrate sollte hierfür das „zulässige“ Verhalten zu überwachender Komponenten dokumentiert sein.

Korrelation der Daten

Der Prozess einer zweckmäßigen Angriffsanalyse setzt die fortwährende, automatische Korrelation einzubeziehender Daten voraus. Dabei bezeichnet die Korrelation hier allgemein die Berücksichtigung mehrerer, nicht zeitgleicher Ereignisse eines einzelnen Sensors oder mehrerer Ereignisse verschiedener Sensoren durch die angewandte Auswertungslogik. Erst durch Korrelation übergreifender Informationen mehrerer Quellen auf höherer Ebene lassen sich beispielsweise verteilte, groß angelegte Angriffe nachvollziehen, die einzelnen Sensoren sonst unauffällig erscheinen. Hinsichtlich der Wirksamkeit in einer mächtigen Umgebung, wie derjenigen von Grids, muss besonders auf den Aspekt der Korrelation über domänenüberschreitende Informationen äußerst heterogener Sensoren eingegangen werden, was beispielsweise durch geeignete Datenaustauschformate vorangebracht werden kann.

Nutzer-Discovery

Neben des Einblicks in Ressourcen und Dienste des Grids ist genauso auch ein permanentes Wissen um den aktuellen Kreis der Nutzer von essenzieller Bedeutung. Ein gridbasiertes Intrusion Detection System muss effektiv dabei unterstützt werden, autorisierte Nutzer samt ihrer jeweiligen Rollenausprägung gezielt abzufragen. Zu diesem Zweck kann wiederum eine Ankopplung an diesbezügliche Managementsysteme förderlich sein.

Pull-/Push-Datenzugriffsmodell

Analysiert man die Anforderungen an Intrusion Detection Systeme im Umfeld von Grids, kommt man nicht umhin, sich auch mit der Frage eines sinnvollen Modells für den Datenzugriff zu beschäftigen. Generell lassen sich vor allem das Pull- (engl. *ziehen*) und das Push-Modell (engl. *schieben*) unterscheiden. In ersterem Fall stellt der Informationsanbieter sämtliche Daten erst nach ausdrücklicher Anfrage bereit, wohingegen für das Push-Modell gilt, dass die Daten vom Informationsanbieter selbsttätig regelmäßig verbreitet werden. Dies kann in konstanten zeitlichen Abständen geschehen oder jeweils durch bestimmte Ereignisse (beispielsweise für den Fall eines erkannten Angriffs) ausgelöst werden. Eine gridweite Einbruchserkennung sollte beide Verfahren wirksam verbinden, um aus deren Kombination den maximalen Nutzen ziehen zu können.

Überwachbarkeit

Die im Grid kooperierenden Institutionen sind häufig längst im Besitz hoch entwickelter, angepasster Monitoringsysteme für die fortwährende Überwachung ihrer verschiedenartigen Komponenten.

Auch für die Einbruchserkennung sollte die Durchführbarkeit eines kontinuierlichen Monitorings vorausgesetzt werden, wobei insbesondere dessen Eingliederung in bestehende Überwachungskonzepte von vordergründiger Relevanz ist. Hierbei ergibt sich als Anforderung an das anzustrebende System grundsätzlich die Verfügbarkeit adäquater, also präziser und reichhaltiger Daten zum aktuellen Zustand des Systems.

Unterstützung einer bedarfsgesteuerten Analyse im Nachhinein

Zusätzlich zu einer zeitnahen Analyse kann auch die Unterstützung einer bedarfsgesteuerten Analyse im Nachhinein als Anforderung an das zu entwickelnde gridbasierte Intrusion Detection System identifiziert werden. Zunächst belanglos erscheinende Ereignisse können manchmal erst innerhalb des Zusammenhangs einer langfristigen Analyse zu signifikanten Ergebnissen führen. Daneben kann eine Analyse im Nachhinein auch für die Ausgabe spezifischer Berichte in Betracht kommen. Voraussetzung einer solchen zeitversetzten Auswertung ist auf jeden Fall eine möglichst umfangreiche und nachhaltige Speicherung aller infrage kommenden Ereignisdaten, samt zugehöriger Kontextinformationen. Zu beachten ist, dass damit ein inne liegendes Konfliktpotential bezüglich der Anforderung an den Datenschutz verbunden ist.

Unterstützung einer hohen Dynamik berechtigter Nutzer

Ebenso wie die Menge der aktuell eingebrachten Ressourcen kann sich hinsichtlich Zusammensetzung und Rollenzuteilung auch die Gruppe derer beispielsweise zur Laufzeit von Projekten verändern, die innerhalb virtueller Organisationen Zugriff auf die vielfältigen Grid-Dienste besitzen. Der Umgang mit sich häufig verändernden Nutzergruppen ist daher eine wichtige Anforderung.

Unterstützung einer hohen Dynamik einzubeziehender Ressourcen

Die Zusammensetzung der Ressourcen eines Grids unterliegt einem fortwährenden Anpassungsprozess, der von vielfältigen Faktoren abhängt. Innerhalb virtueller Organisationen werden bedarfsgerecht weitere Ressourcen hinzugefügt und nicht länger benötigte entfernt. Betreffende Organisationen entscheiden stets selbst darüber, wann sie welche und wie viele Ressourcen einbringen wollen. Die Einbruchserkennung muss daher prinzipiell eine hohe Dynamik einzubeziehender Ressourcen beachten.

Verschiedene Granularitätsstufen

Aufgrund der Mächtigkeit des anzunehmenden Einsatzszenarios erscheint es dringend angebracht, aus Managementsicht die Gesamtmenge an verfügbaren Informationen aufzugliedern, indem verschiedene Granularitätsstufen entweder zur Wahl gestellt oder von vornherein dem jeweiligen Kontext zugeordnet werden. So kann einerseits eine gröber aufgelöste Ansicht einem raschen Überblick über das System dienen, wohingegen feinere Ansichten nur noch die gerade benötigten Informationen zur Darstellung bringen.

Wartbarkeit/Dokumentation

Ein Höchstmaß an Wartbarkeit sollte für jede hoch entwickelte, komplexe Software erstrebenswert sein, um die spätere Durchführbarkeit von Änderungen bestmöglich zu unterstützen. Als wichtiges Kriterium hierfür gilt die Qualität der Dokumentation zum Produkt, wobei prinzipiell die exakte Spezifikation möglichst aller Schnittstellen von Belang ist. Hilfreich sollte gleichfalls ein modularer, gegliederter Systemaufbau sein.

4.5 Zusammenfassung

Im Zusammenhang mit der Entwicklung einer neuartigen, gridbasierten Einbruchserkennung wendet sich das vorausgegangene Kapitel der Analyse zur Gewinnung von wichtigen generischen Anforderungen an diesbezügliche Systeme zu.

Vorneweg erläutert Abschnitt 4.1 die für die im weiteren durchzuführende Anforderungsanalyse gewählte Vorgehensweise. Es ist hierbei entscheidend, dass sich die anzustrebende übergreifende Lösung bestmöglich in ein komplexes Umfeld integrieren muss. Es erscheint einleuchtend, dass dazu größter Wert auf ein wohl überlegtes Vorgehen gelegt werden muss. Damit eine Applikabilität des Verfahrens in Bezug auf potenzielle Zielumgebungen gewährleistet werden kann, werden hohe Ansprüche an die Analyse gestellt.

Abschnitt 4.2 widmet sich der Identifikation beteiligter Aktoren. Die Komplexität der Erkennung von Angriffen im Grid führt zu einer Vielzahl an abzugrenzenden Aktoren, die jeweils kurz beschrieben werden. Neben personellen Aktoren müssen gleichfalls weitere ergänzende, vor allem technischer Art in Betracht gezogen werden.

Als wesentlicher Bestandteil des Kapitels werden in Abschnitt 4.3 relevante Anwendungsfälle zusammengestellt, die dann zur Ermittlung der jeweiligen Anforderungen dienen. Zur Strukturierung wurde hierbei eine Aufteilung in mehrere Dimensionen vorgenommen, die ein möglichst weit greifendes und vollständiges Bild der maßgeblichen Anforderungen an gridweite Intrusion Detection Systeme bieten. Im einzelnen umfasst dieses Vorgehen Anwendungsfälle bezüglich der organisationsübergreifenden Zusammenarbeit im Grid, der Einbruchserkennung spezifisch in Grids, der Integration in übrige Systeme sowie weiterer grundlegender Aspekte. Auf diese Weise werden insgesamt 48 verschiedenen Anforderungen abgeleitet. Abschließend werden die Anforderungen und die Anwendungsfälle tabellarisch zusammengestellt, um so die Beurteilung der Relevanz einzelner Anforderungen zu unterstützen. Mit zu den wichtigsten Anforderungen zählen demnach unter anderem die Interoperabilität sowie die Portabilität und Wiederverwendbarkeit, einheitliche Schnittstellen oder auch die Wahrung der Integrität.

Abschnitt 4.4 bietet jeweils eine kurze Beschreibung sowie Erläuterungen zur konkreten Relevanz betreffend den gesamten Katalog der zuvor erarbeiteten Anforderungen.

KAPITEL FÜNF

Klassifizierung von Angriffen auf Grids

Inhalt des Kapitels

5.1 Bedrohungsanalyse: Disziplin des Security Engineering 74
5.2 Frühere Arbeiten 74
5.2.1 Dreigliedriges Schema nach Gerloni et al. [GORP04] 75
5.2.2 Bedrohungsmatrix und -baum nach Eckert [Eck06] 75
5.2.3 Grid-Angriffe nach Schulter et al. [SNKW06] 76
5.2.4 Weitere Ansätze 77
5.2.5 Vergleich der Konzepte 77
5.3 Klassifikation der Angreifer 78
5.4 Klassifikation der Angriffe 80
5.4.1 Bedrohungen durch externe Angriffe 80
5.4.2 Bedrohungen der Datenintegrität und der Informationsvertraulichkeit . . 81
5.4.3 Bedrohungen der Verfügbarkeit und der Ressourcennutzung 81
5.5 Zusammenfassung 82

Einbruchsversuche in Grids müssen wohl mit dem weiteren Anstieg der Leistungsfähigkeit immer mehr als besonders attraktiv und so geradezu als „magische“ Anziehungspunkte aufgefasst werden, was in der Folge gleichfalls eine entsprechend hohe darauf gerichtete Aktivität erwarten lässt (vgl. Abschnitt 1.1). Nach der substantiierten Ermittlung wichtiger generischer Anforderungen an grid-basierte Intrusion Detection Systeme des vorherigen Kapitels ist es ebenso noch Ziel dieser Arbeit, hinsichtlich einer wirksamen Einbruchserkennung im Grid die möglichen Angriffe im Rahmen einer grundlegenden Bedrohungsanalyse systematisch zu erfassen.

Einleitend motiviert Abschnitt 5.1 das Kapitel. Es wird hierbei die wichtige Problemstellung der Bedrohungsanalyse allgemein in den übergeordneten Kontext des „Security Engineering“ eingeordnet, das sämtliche relevanten Maßnahmen und Methoden zur Konstruktion sicherer Systeme vereint.

Als Grundlage für die im weiteren anzustrebende Klassifikation ist es förderlich, sich zunächst mit früheren Arbeiten auf diesem Gebiet auseinanderzusetzen. Abschnitt 5.2 fasst die wichtigsten davon kompakt zusammen, wobei die Konzepte zuletzt kurz verglichen und ihre etwaige Übertragbarkeit auf das Grid beurteilt wird.

Abschnitt 5.3 stellt schließlich die verschiedenen Typen potenzieller Angreifer zusammen. Im Anschluss daran widmet sich Abschnitt 5.4 der Entwicklung einer möglichst umfassenden und robusten Klassifikation der Angriffe, denen Grids potenziell ausgesetzt sind. Wie bereits in Abschnitt 4.4 erklärt, sieht man sich neben der Bedrohung mit herkömmlichen und hinreichend bekannten, genauso auch in weniger umfangreichen Umgebungen auftretenden Angriffstypen zunehmend mit einem neuen, gridspezifischen Angriffspotenzial konfrontiert.

5.1 Bedrohungsanalyse: Disziplin des Security Engineering

Die Entwicklung sicherer Systeme stellt offensichtlich eine große Herausforderung dar. In jüngerer Zeit fanden daher Entwicklungsprozesse zunehmend Beachtung, die versuchen, sichere Systeme wohl strukturiert zu konstruieren. Das sogenannte „Security Engineering“ lehnt sich an allgemeine, methodische und in der Regel top-down orientierte Vorgehensmodelle des Software-Engineerings an, da speziell auf das Security Engineering ausgerichtete Verfahren bislang kaum entwickelt wurden (vgl. [Eck06, GORP04]). Neben der diesem Kapitel zuzuordnenden *Bedrohungsanalyse* zählen hierzu noch weitere Schritte und Phasen, die nachfolgend jeweils knapp beschrieben werden. Dies betrifft hauptsächlich die *Strukturanalyse*, die *Schutzbedarfsermittlung*, die *Risikoanalyse* sowie die Definition einer *Sicherheitsstrategie*.

Da sicherheitskritische Anforderungen zumeist eng mit der geplanten Zielumgebung des Systems verknüpft sind, werden in der Strukturanalyse initial die vorhandenen Komponenten der Infrastruktur in Form einer Bestandsaufnahme beschrieben. Unter Umständen kann die Strukturanalyse auch um etwaige weitere benötigte Komponenten ergänzt werden, sofern diese für die Betrachtung im Einzelfall von Relevanz sind.

Für die Festlegung von Zielen betreffend die Sicherheit gilt es, zuerst den konkreten Schutzbedarf im Hinblick auf die Vertraulichkeit, Integrität und Verfügbarkeit festzustellen, der sich entweder aus der individuellen Gefährdungslage oder auch aus typischen Schadensszenarien ableiten lässt. Ist der Schutzbedarf bekannt, müssen die Defizite zwischen dem gegenwärtigen und dem Sollzustand gefunden werden.

Die Klassifizierung von Angriffen auf Grids muss zur Bedrohungsanalyse gezählt werden, die sich allgemein der Erhebung von potenziellen organisatorischen, technischen und benutzerbedingten Ursachen für relevante Bedrohungen widmet. Hierbei muss größter Wert auf ein systematisches Vorgehen gelegt werden, um alle einzubeziehenden Bedrohungen möglichst umfassend abzubilden.

Im Anschluss an die Bedrohungsanalyse sollte gleichfalls eine Risikoanalyse zur Bewertung der Bedrohungen durchgeführt werden. Es wird versucht, für jede Bedrohung eine zugehörige Eintrittswahrscheinlichkeit zu ermitteln, sowie den potenziell einhergehenden Schaden abzuschätzen. Aus dem Produkt lässt sich das quantitative Risiko berechnen, das sich besonders für eine Priorisierung umzusetzender Sicherheitsmechanismen eignet. Hinsichtlich der Schadenshöhe werden primäre und sekundäre Schäden unterschieden. Leichter zu quantifizierende primäre Schäden stellen beispielsweise Schäden bedingt durch einen Produktivitätsausfall, Wiederbeschaffungs-, Personal- oder Wiederherstellungskosten dar. Zu den sekundären Schäden werden ein möglicher Imageverlust oder auch ein Vertrauensverlust bei den Kunden gezählt.

Die zur Umsetzung des Schutzbedarfs erforderlichen Maßnahmen sind in einer Sicherheitsstrategie (engl. *Security Policy*) informell oder besser präzise formalisiert zu erfassen.

Generell gilt, dass das Security Engineering nicht als einmalige Aktivität anzusehen ist. Vielmehr sollten die einzelnen Teilschritte immer wieder durchlaufen werden, um dauerhaft ein hohes Sicherheitsniveau sicherzustellen.

5.2 Frühere Arbeiten

Im Laufe der Jahre wurden eine Reihe von Ansätzen zu Angriffsklassifikationen publiziert, die mitunter breite Anerkennung fanden. Dazu gehören unter anderem das bekannte dreigliedrige Schema nach Gerloni et al. [GORP04], die Bedrohungsmatrix und der Bedrohungsbaum nach Eckert [Eck06],

die im Rahmen der vorliegenden Arbeit spannende, weil gridbezogene Spezifikation nach Schulter et al. [SNKW06] sowie einige weitere Ansätze.

Die folgenden Abschnitte behandeln jedes der angegebenen Schemata in der gebotenen Kürze, wobei die angegebenen Quellen dem interessierten Leser jedoch die Möglichkeit zur detaillierteren Begutachtung geben. Zuletzt werden die Konzepte verglichen und hinsichtlich ihrer Übertragbarkeit auf Grid-Szenarien beurteilt.

5.2.1 Dreigliedriges Schema nach Gerloni et al. [GORP04]

Abgesehen von der geläufigen Klassifikation nach internen und externen Angriffen unterscheiden Gerloni et al. aktive Angriffe, Denial-of-Service (DoS) und autonome Angriffe.

Aktive Angriffe bezeichnen dabei vorsätzliche und kontrolliert auf ein bestimmtes System ausgeführte Angriffe, wobei der Angreifer im Erfolgsfall in die Lage versetzt wird, Daten auszuspähen und auch zu verändern. Es muss hierbei sowohl von externen wie auch von internen Angreifern ausgegangen werden. Als Beispiele für diese Art von Angriffen werden neben dem sogenannten „Session Hijacking“ bei dem Angreifer bestehende, bereits autorisierte Verbindungen übernehmen, insbesondere das DNS- und IP-Spoofing genannt.

Eine DoS-Attacke dient dem Blockieren bestimmter Dienste, wobei viele dieser Angriffe auf der Herbeiführung einer simplen Überlastung beruhen. Werden Angriffe unter missbräuchlicher Nutzung einer größeren Anzahl an kompromittierten Systemen durchgeführt, spricht man von „Distributed Denial-of-Service“ (DDoS). Da Angreifer zudem häufig Fehler der jeweils eingesetzten Software ausnutzen, werden regelmäßig keine besonderen Zugriffsrechte vorausgesetzt. Bei einer SYN-Flood-Attacke werden eine übermäßige Zahl halboffener Verbindungen aufgebaut, um den Zielrechner an der Annahme weiterer Verbindungen zu hindern. Grundsätzlich sind weitere DoS-Szenarien denkbar.

Als dritte Gruppe möglicher Angriffe wird auf autonome Angriffe hingewiesen, die von eigenständigen Programmen oder Programmteilen ausgeführt werden. Diesen ist es möglich, selbstständig und unkontrollierbar weitere Systeme anzugreifen. Um Autonome Angriffe systematisch einzuordnen, werden diese in Viren, Würmer und Trojaner eingeteilt. Ein Virus ist dabei kein selbstständig ausführbares Programm, sondern eine Befehlsfolge, die ein Wirtsprogramm zu seiner Ausführung benötigt. Ein Wurm ist dagegen selbst ein ausführbares Programm, das ausdrücklich keinen Wirt mehr benötigt. In beiden Fällen wird versucht, sich durch Selbstreplikation weiter zu verbreiten, wodurch weitere Schäden durch Überlastung von Netzen oder Rechnern herbeigeführt werden können. Als trojanisches Pferd (Trojaner) werden schließlich Programme bezeichnet, dessen wahre Funktion nicht mit der vorgespiegelten Funktion übereinstimmt.

5.2.2 Bedrohungsmatrix und -baum nach Eckert [Eck06]

Im Sinne eines wohl überlegten, methodischen Vorgehens zur Erfassung sämtlicher relevanter Bedrohungen bieten sich nach Eckert im Großen und Ganzen zwei Verfahren an, nämlich ein matrix- und ein baumorientierter Analyseansatz.

Im Falle der Bedrohungsmatrix klassifiziert man anfangs die einzelnen Gefährdungsbereiche, die sodann die Zeilen der Matrix bilden. Zur Abdeckung der wichtigsten Problembereiche wird eine Aufspaltung der Gefährdungsbereiche in fünf Gruppen vorgeschlagen:

1. Bedrohungen durch externe Angriffe
2. Bedrohungen der Datenintegrität und der Informationsvertraulichkeit

3. Bedrohungen der Verfügbarkeit und der Ressourcennutzung
4. Abstreiten durchgeführter Aktionen
5. Missbrauch erteilter Berechtigungen

Die Spalten der Matrix werden durch die potenziellen Angreifer definiert. In die Matrix werden zuletzt die potenziellen Angriffsszenarien eingetragen.

Als alternatives Verfahren für eine möglichst substantiierte Erhebung relevanter Bedrohungen werden Bedrohungs- bzw. Angriffsbäume (engl. *Attack Tree*) angeführt. Die Wurzel eines Bedrohungsbaumes definiert ein mögliches Angriffsziel und damit eine anzunehmende Bedrohung des Systems. In der nächsten Ebene des Baumes werden Zwischenziele formuliert, die zur Erreichung des Gesamtzieles beitragen. Als Blätter des Baumes fungieren sodann die einzelnen Angriffsschritte, wobei nun die unterschiedlichen Pfade von den Blättern zur Wurzel die möglichen Wege zum Erreichen des globalen Angriffsziels an der Wurzel beschreiben. An hierfür in Betracht zu ziehenden, allgemeinen Angriffszielen werden unter anderem das Vortäuschen der Identität eines autorisierten Benutzers (Maskierungsangriff) oder auch der unautorisierte lesende oder schreibende Zugriff auf gespeicherte Daten und Informationen angegeben.

Bedrohungsbäume sind auch im Hinblick auf eine sich anschließende Risikoanalyse vorteilhaft, indem die Risiken der möglichen Bedrohungen als Attributierungen der Kanten des Baumes angegeben werden können.

5.2.3 Grid-Angriffe nach Schulter et al. [SNKW06]

Im Gegensatz zu den vorherigen, generischen Klassifizierungsansätzen verfolgen Schulter et al. das Ziel, speziell die im Rahmen von Grids relevanten Angriffsmuster erschöpfend in ein geeignetes Schema einzugliedern. Sie schlagen vier sich komplettierende Klassen vor, nämlich den unerlaubten Zugriff auf Ressourcen, die missbräuchliche Nutzung von Ressourcen, Exploits sowie zusätzlich noch host- oder netzspezifische Angriffe. Im folgenden sollen diese kurz beschrieben werden.

Unter einem unerlaubten Zugriff auf Ressourcen wird ein Einbruch verstanden, bei dem sich der Angreifer als legitimer Benutzer des Grids maskiert. Hierfür notwendige Passwörter können durch Ausspähen, mittels eines Brute-Force-Angriffs, durch Erraten oder durch den leichtfertigen Umgang derselben durch berechtigte Nutzer erlangt werden. Weiterhin muss möglicherweise auch ein Angriff auf den Authentisierungsdienst selbst in Betracht gezogen werden.

Die missbräuchliche Nutzung von Ressourcen folgt entweder aus dem vorgenannten Fall eines unerlaubten Zugriffs oder aber aus einer Überschreitung der Rechte zugelassener, interner Nutzer. Eine entsprechende gridbezogene Sicherheitsrichtlinie sollte hierzu ein abweichendes Nutzerverhalten genau spezifizieren, wozu etwa eine außerordentlich starke oder böswillige Beanspruchung von Ressourcen sowie unbeabsichtigte Fehler von Benutzern zählen sollten.

Die als weitere Gruppe aufgefassten Exploits beziehen sich auf Angriffe, die unter Zuhilfenahme von besonderen Werkzeugen gezielt Schwachstellen ausnutzen, die bisweilen in Protokollen, Diensten und Anwendungen des Grids offenstehen. Oftmals trifft dies auf diverse als Denial-of-Service ausgeführte Angriffe oder Würmer zu, die ihre Spuren an vielerlei Orten der Infrastruktur hinterlassen.

Der letzten Rubrik der host- oder netzspezifischen Angriffe ordnen die Autoren sämtliche übrigen Angriffe zu, die typisch für einzelne Hosts oder Netze sind, ohne darauf jedoch detaillierter einzugehen. Vorwiegend soll hierdurch wohl der Anspruch auf Vollständigkeit der vorgenommenen

Klassifizierung bekräftigt werden.

5.2.4 Weitere Ansätze

Auf dem Gebiet der Untersuchung von Angriffen wurden immer wieder interessante Arbeiten vorgestellt, die zum Teil schon viele Jahre zurückreichen und trotzdem immer noch aktuell sind. Die wichtigeren dieser Konzepte werden analog zu [LJ97]) in den nächsten Absätzen wiedergegeben.

Ein früher Vorschlag geht auf Lackey [Lac74] zurück, der sechs verschiedene Kategorien relevanter Einbruchstechniken zusammenfasste, ohne aber seine Aussagen durch konkrete, geeignete Referenzen zu untermauern.

Die Konzeption des Schemas von Neumann und Parker [NP89] liegt nun schon einige Zeit zurück, wobei es zwischenzeitlich mehrfach von anderen Autoren aufgegriffen und weiter verfeinert wurde. Es beinhaltet eine Kategorisierung von Angriffen in neun Gruppen, die die meisten bekannten Typen abdecken. Als Datenbasis dienten den beiden mehr als 3000 Angriffe, die über einen Zeitraum von 20 Jahren beobachtet werden konnten. Die Verfasser betonen jedoch, dass sich die einzelnen vorgestellten Rubriken nicht gegenseitig ausschließen, wodurch manche Angriffsszenarien gegebenenfalls Techniken mehrerer dieser Klassen einschließen. Der Ansatz zeichnet sich durch eine wohl strukturierte Anordnung der Klassen aus, von physischen Gegebenheiten (Klasse 1) zur Hardware (Klasse 2) und weiter zur Software (ab Klasse 3), und außerdem von nichterlaubter Nutzung zum Missbrauch von Befugnissen.

Daneben erlangte die von Brinkley und Schell [BS95] herausgebrachte Gegenüberstellung der verschiedenen Angriffe einige Bekanntheit. Sie erschufen sechs sich ebenfalls nicht ausschließende Klassen für den sogenannten informationsorientierten Missbrauch, der die sicherheitsrelevanten Eigenschaften der Informationsvertraulichkeit und der Integrität betrifft. Der ressourcenorientierte Missbrauch hinsichtlich der Verfügbarkeit ging nicht mitein. Die Arbeit erscheint nichtsdestotrotz abgesehen von einer kleinen Anzahl von zitierten Beispielen nur wenig fundiert.

5.2.5 Vergleich der Konzepte

Generell gibt es verschiedene Möglichkeiten (sogenannte *Dimensionen*), Angriffe effektiv in Kategorien einzuordnen (vgl. [LJ97])). Infrage kommen beispielsweise die Klassifizierung nach angegriffenen Komponenten, nach der Absicht des Angreifers, nach den zur Ausführung von Angriffen benutzten Techniken, nach ausgenutzten Schwachstellen oder auch nach der tatsächlich erzielten Wirkung von Angriffen (wie etwa dem Verlust der Informationsvertraulichkeit).

Die zuvor angeführten Ansätze sind alle relativ ähnlich und verfolgen zumeist die Klassifizierung nach den Angriffstechniken, wobei demgegenüber etwa das Schema von Eckert (vgl. Abschnitt 5.2.2) hauptsächlich auf die Wirkung von Angriffen abzielt. Unterschiede ergeben sich allenfalls in der Granularität der Klassen, wobei zumeist nur wenige davon aufgeführt werden. Das Schema von Neumann und Parker (vgl. Abschnitt 5.2.4) erscheint hierbei mit neun verschiedenen Rubriken besonders ausgeprägt gegliedert. Interessant ist in diesem Zusammenhang, dass sich die vorgeschlagenen Klassen oftmals nicht gegenseitig ausschließen und einzelne Angriffe demzufolge zu mehreren Klassen gehören können.

Da Grids prinzipiell eine Weiterentwicklung bekannter Ideen des verteilten Rechnens darstellen, sind bisherige Klassifizierungen von Angriffen grundsätzlich auch übertragbar. Gegebenenfalls werden Anpassungen erforderlich, die die Spezifika von Grids individuell betrachten. Grids unterscheiden

sich von herkömmlichen Nicht-Grid-Umgebungen vor allem durch die potenziell größere Leistungsfähigkeit und die möglichen Schäden; in Grids ist insgesamt von einer deutlich verschärften Gefährdungslage auszugehen. Der effiziente und automatisierte Einsatz einer potenziell großen Menge an Ressourcen, die große Gemeinde der Nutzer und der effiziente und automatisierte Einsatz von Ressourcen kann von Angreifern zu ihrem Vorteil genutzt werden [SNKW06].

5.3 Klassifikation der Angreifer

Vor einer Untersuchung expliziter Angriffe ist es im Sinne einer strukturierten, die verschiedenen Gefährdungslagen in angemessener Weise erfassenden Bedrohungsanalyse hilfreich, zunächst die potenziellen Angreifer genau zu analysieren. Da Grids substanzielle Parallelen zu herkömmlichen verteilten Umgebungen aufweisen, sollten diesbezüglich bereits anerkannte Konzepte größtenteils übertragbar sein.

Gerloni et al. [GORP04] beschreiben ein Angreifermodell (vgl. Abbildung 5.1), um infrage kommende Angreifer nach ihrer Gefährlichkeit einordnen zu können. Dabei wird jeder Angreifer nach Know-how sowie seinem möglichen Zeit- und Ressourceneinsatz eingeteilt.

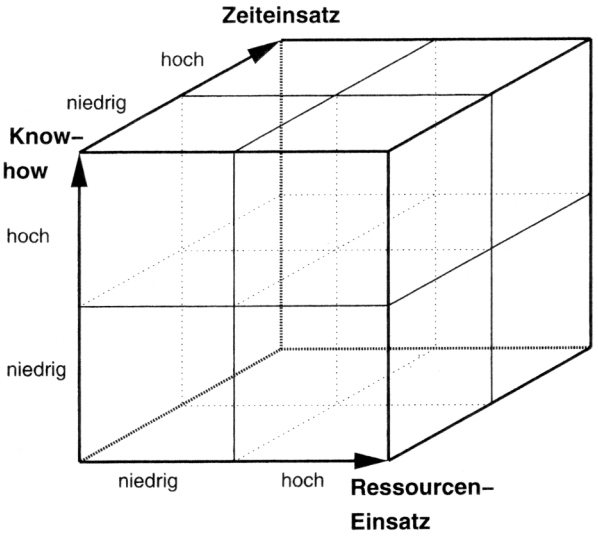

Abbildung 5.1: Angreifermodell [GORP04]

Grundsätzlich sind Grids einer Vielzahl an vorstellbaren Angreifern ausgesetzt. Sie lassen sich im Wesentlichen den nachfolgenden Klassen zuordnen [Eck06, GORP04]:

Hacker. Ursprünglich verstand man unter einem Hacker den technisch äußerst versierten Administrator, dem es möglich war, Fehler durch geschicktes Vorgehen (engl. *Hack*) für eigene Zwecke zu missbrauchen. Etwas allgemeiner gefasst ist es also vorrangiges Ziel des Hackers, Schwachstellen in IT-Systemen zunächst aufzudecken und mittels dedizierter Angriffe (sogenannte *Exploits*) schließlich auch auszunutzen.
Es sei jedoch darauf hingewiesen, dass Hacker in aller Regel „anständige“ Ziele verfolgen, indem sie ihre Erkenntnisse veröffentlichen, um auf eklatante Schwachstellen hinzuweisen und um somit vor allem die Entwicklung sicherer Systeme voranzubringen. Hackern liegt insbesondere nicht an einer persönlichen Bereicherung (etwa in Form finanzieller Gewinne). Der hierfür geprägte Begriff der „Hacker-Ethik“ soll jedoch nicht darüber hinwegtäuschen, dass Hacker zur Durchführung ihrer Angriffe in vielen Fällen illegale Wege beschreiten.

Cracker. Unter einem Cracker ist demgegenüber ein böswilliger Angreifer zu verstehen. Wie ein Hacker stellt er einen ebenso qualifizierten Angreifer dar, wobei er Angriffe allerdings gezielt zu seinem eigenen Vorteil oder auch zum Nachteil eines Dritten durchführt.
Verglichen mit Hackern geht somit von Crackern ein weitaus größeres Bedrohungspotenzial aus. Da Grids zunehmend ein äußerst lohnendes Ziel für Angriffe abgeben, sollte größtmöglicher Aufwand betrieben werden, um Angriffe durch Cracker nach Möglichkeit vollends abzuwehren oder wenigstens das Schadensausmaß zu begrenzen.

Skript Kiddies. Im Gegensatz zu Hackern oder Crackern verfügen Skript Kiddies zwar nicht notwendigerweise über tief greifendes technisches Know-how, dafür aber über viel Zeit.
Stattdessen greifen sie auf vorhandene Werkzeuge und frei verfügbare Exploits zurück, um vorwiegend aus Neugierde oder um Aufmerksamkeit auf sich zu lenken, Angriffe auszuführen. Auch wenn Skript Kiddies sicherlich primär nicht das Ziel verfolgen, Dritte vorsätzlich zu schädigen, geht von Ihnen doch schon aufgrund ihrer schieren Masse eine erhebliche Gefahr aus.

Staaten und politische Organisationen. Es ist offensichtlich, dass Staaten und politische Organisationen über außerordentliche Mittel und Wege verfügen, um jedwede Angriffe wirksam durchzuführen. Insbesondere stehen solchen Stellen erhebliche personelle und technische Ressourcen zur Verfügung.
Dies erfolgt vorwiegend zum Zwecke der Wirtschaftsspionage oder zur Durchsetzung geheimdienstlicher Ziele. Auch unter Berücksichtigung einer weiter ansteigenden Verbreitung der industriellen Nutzung von Grids sollte diesem großen Bedrohungspotenzial eine entsprechende Bedeutung und Aufmerksamkeit beigemessen werden.

Kriminelle. Zusätzlich zu Staaten oder politischen Organisationen müssen auch andere Gruppierungen für schwerwiegende, kriminelle Angriffe auf vernetzte Systeme in Betracht gezogen werden. Eine organisierte Kriminalität kann unter anderem von terroristischen Vereinigungen ausgehen.
Dabei wird hauptsächlich auf bestimmte, eng begrenzte Zielgruppen (Regierungsstellen, Banken etc.) fokussiert, um jeweils einen möglichst großen Schaden anzurichten. Charakteristisch für diese Art von Angriffen ist insbesondere, dass hierfür zunehmend technisches Know-how von entsprechend versierten Angreifern „hinzugekauft“ wird.

Mitarbeiter. Gerade auch im Kontext von möglicherweise sehr umfänglichen Gridverbünden mit einer schwer zu überblickenden Zahl von berechtigten Nutzern geht eine besonders große Gefahr von Angreifern aus den eigenen Reihen aus, weil sie bereits über eine mehr oder weniger ausgeprägte Kenntnis relevanter Abläufe und der Infrastruktur verfügen (sogenanntes *Insider-Wissen*).
Ihre Motive liegen unter Umständen im finanziellen Anreiz, oder aber auch in Rache. Interne Angreifer müssen oftmals keine speziellen Sicherheitsbarrieren (Firewalls etc.) überwinden, da sie zur Erfüllung ihrer Tätigkeit ohnehin mit gewissen Zugriffsrechten auf die Infrastruktur ausgestattet sind. Mit Maßnahmen eines ganzheitlichen Sicherheitsmanagements könnten etwaige bestehende organisatorische oder technische Sicherheitsmängel effektiv behoben werden.

Verschiedene Studien (siehe hierzu u.a. [Ric03]) haben zu dem aufschlussreichen Ergebnis geführt, dass ein Großteil beobachteter Angriffe in verbreiteten Umgebungen (Unternehmen, Firmen, Behörden) auf die internen Mitarbeiter selbst zurückgeführt werden kann. Trotzdem erfahren Angriffe durch Hacker oder Cracker in der Öffentlichkeit wesentlich mehr Beachtung, obwohl sie bislang doch eher selten vorkommen und zumeist nur geringere Schäden verursachen. Ein in vielerlei Hinsicht offenes System, wie es gerade Grids verkörpern, lässt jetzt aber ein erheblich gesteigertes Be-

drohungspotenzial durch solche externen Angreifer erwarten. [Eck06]

5.4 Klassifikation der Angriffe

Die Erfassung von gridbezogenen Bedrohungen stellt eine große Herausforderung dar, wobei insbesondere fundierte Kenntnisse über etwaige infrage kommende Sicherheitsprobleme im Kontext von Grids vorauszusetzen sind. Gemäß Amoroso [Amo94] sollte jedwede Klassifizierung von Angriffen insbesondere die folgenden Eigenschaften erfüllen:

Vollständigkeit. Eine Taxonomie sollte möglichst vollständig in Bezug auf die Abdeckung von Angriffen sein.

Richtigkeit. Die ausgewählte Taxonomie sollte alle infrage kommenden Angriffe adäquat charakterisieren. Hierfür maßgebliche Randbedingungen sollten vor der Anwendung spezifiziert werden.

Einbeziehung von internen und externen Bedrohungen. Grundsätzlich sollte eine Taxonomie Angriffe von internen und von externen Angreifern berücksichtigen.

Der in der vorliegenden Arbeit einzunehmende Fokus liegt ausdrücklich nicht auf der Untersuchung einzelner Sicherheitslücken (engl. *Bugs*), von denen angenommen werden muss, dass sie in Protokollen, Diensten und Anwendungen des Grids (zum Beispiel in verbreiteten Middlewares wie dem Globus Toolkit, in Betriebssystemen etc.) bestehen. Dies folgt einerseits aus der allgemeinen Komplexität und damit verbundenen zahlreichen Abhängigkeiten. Außerdem lassen sich kaum Hinweise auf disziplinierte Softwareentwicklungsprozesse auffinden, die besonders auch das Security Engineering verantwortlich und umfassend umsetzen. Generell unterliegen die im Grid verbundenen Systeme ebenfalls vielen typischen Bedrohungen, die für herkömmliche Rechner und die sie verbindenden Netze gelten (vgl. [Ken99]).

Zur Unterstützung eines wohl überlegten und substantiierten Vorgehens bietet es sich an, gerade im Kontext von Grids die Klassifikation der Angriffe an die weithin anerkannten Gefährdungsbereiche nach Eckert (vgl. Abschnitt 5.2.2) anzulehnen. Insbesondere erscheint dieses Vorgehen vor dem Hintergrund einer möglichst allgemeingültigen Klassifikation auf hohem Abstraktionsgrad vorteilhaft, da grundsätzlich von der elementaren Definition eines Angriffs als diejenige die Integrität, Vertraulichkeit oder Verfügbarkeit kompromittierende Aktivität ausgegangen wird. In den folgenden Abschnitten werden die Gefährdungsbereiche dazu fallweise für das spezifische Umfeld des Grids instantiiert, wobei vor allem auch geeignete Beispiele für Angriffe gegeben werden.

5.4.1 Bedrohungen durch externe Angriffe

Genauso wie viele herkömmliche Umgebungen sind auch Grids von externen Angriffen bedroht, die sich durch Aktionen eines Angreifers ergeben, die er ohne Hilfe des bedrohten technischen Systems durchführt.

Hierzu zählen in erster Linie Vandalismus im Sinne einer physischen Zerstörung oder Entwendung, beispielsweise von wichtigen Systemkomponenten, wie Servern oder Netzverbindungen, von Datenträgern, wie Festplatten, oder ebenso sekundäre Bedrohungen, die sich unter anderem aus der Unterbrechung von Versorgungsleitungen ergeben können. Auch ein visuelles Ausspähen von Passwörtern wird zu dieser Gruppe gerechnet.

5.4.2 Bedrohungen der Datenintegrität und der Informationsvertraulichkeit

In diesen Problembereich fallen Angriffe, die gezielt versuchen, interne Kontrollen zu umgehen oder Schutzmechanismen unwirksam zu machen, um entweder Daten des Grids zu verändern oder um auch nur an vertrauliche und häufig sogar wertvolle Informationen zu gelangen. Hierfür ist ein breites Feld an anzunehmenden aktiven und autonomen Angriffen denkbar, von denen einige Vertreter nachfolgend betrachtet werden.

Um beispielsweise unautorisierten Zugriff auf die Daten zu erhalten, kann sich ein externer Angreifer als legitimer Nutzer ausgeben. An das hierfür notwendige Passwort (beziehungsweise auch an einen etwaigen privaten Schlüssel) gelangt er entweder durch Diebstahl (zum Beispiel durch Belauschen von Verbindungen), mittels eines Brute-Force-Angriffs, durch Erraten oder gegebenenfalls durch wenig sorgfältigen Umgang von berechtigten Nutzern damit. Infrage kommt auch eine generell mängelbehaftete Authentifikation im Grid oder der gezielte Angriff zum Ziele einer Manipulation oder Ausschaltung der Infrastruktur zur Authentifikation und Autorisierung (AAI) der Middleware selbst. Die automatisierte Delegation von Berechtigungen im Grid kann dabei unter Umständen zu einem beträchtlichen Aktionsradius des Angreifers führen.

Daneben sind Angriffe vorstellbar, die bereits autorisierte Verbindungen übernehmen. In diesem Zusammenhang muss auch auf diverse Spoofing-Techniken hingewiesen werden, wie etwa die Manipulation privilegierter Programme für die Erlangung höherwertiger Rechte.

Interne Angreifer verfügen in der Regel bereits über individuelle Berechtigungen, die missbräuchlich genutzt werden können.

Als autonome Angriffe werden schließlich solche Angriffe aufgefasst, die von eigenständigen Programmen oder Programmteilen ausgeführt oder begünstigt werden. Hierzu werden Viren, Würmer oder trojanische Pferde gerechnet.

5.4.3 Bedrohungen der Verfügbarkeit und der Ressourcennutzung

Bedrohungen dieses Gefährdungsbereichs betreffen vornehmlich die Blockade bestimmter Dienste und die unautorisierte Nutzung von Ressourcen des Grids. Der Fall der Blockade wiegt besonders schwer, da hierdurch mitunter wichtige Prozesse unterbrochen werden. Die unautorisierte Nutzung von Ressourcen (Rechen-, Speicherkapazitäten etc.), beziehungsweise die eine „normale“ Ressourcenbeanspruchung übersteigende Verwendung von Ressourcen kann ebenfalls ernste Gefährdungen bewirken.

Um bestimmte Dienste etwa durch einen Denial-of-Service-Angriff (DoS) zu blockieren, benötigt ein Angreifer regelmäßig keine besonderen Rechte auf dem Zielsystem. Neben der Ausnutzung potenzieller Sicherheitslücken in den im Kontext komplexer Grids enthaltenen zahlreichen Protokollen, Diensten und Anwendungen basieren viele dieser Attacken auf einer simplen Überlastung von hierfür disponierten Systemen. Besonders bedroht sind dabei tendenziell sensible und exponierte Komponenten des Grids, wie unter anderem Gateways zwischen den Netzen, oder auch zentrale Aufgaben übernehmende Komponenten (zum Beispiel zur Authentifikation und Autorisierung, für VO-Management, Accounting, Billing oder übrige Managementaufgaben). Solche Angriffe können sich selektiv auf einzelne Systeme beziehen, oder aber auch auf mehrere, sogar domänen- und organisationsübergreifend zugleich. Erfolgen die Angriffe von mehreren, verteilten Systemen aus, wie oftmals gerade bei Wurmangriffen, spricht man von einem verteilten (engl. *Distributed*) Denial-of-Service-Angriff (DDoS).

Ein Ausfall von Ressourcen kann überdies beispielshalber durch Manipulation von Routinginformationen oder der kritischen Namensauflösung erreicht werden.

Die unerlaubte Ressourcennutzung folgt zumeist aus einem unautorisierten Zugriff oder aber aus dem Missbrauch eines legitimen Benutzers und kann unter Umständen durch ein auffällig abweichendes Nutzerverhalten festgestellt werden. Sie lässt sich nach aktiv und passiv differenzieren. Im Falle einer aktiven unerlaubten Nutzung zielt der Angreifer auf die Verwendung der potenziell leistungsfähigen Grid-Ressourcen für eigene Zwecke oder zum Schaden Dritter; hierfür kommen neben monetären Gesichtspunkten bei der Ausführung profitabler Rechenjobs etwa ein böswilliger Einsatz zum Brechen kryptografischer Schlüssel oder zur Durchführung breit angelegter, verteilter Angriffe auf weitere Systeme in Betracht [NR05]. Daneben kann ein übermäßiger Anstieg der Ressourcennutzung die legitime Verwendung von Ressourcen effektiv behindern oder sogar ausschließen. Eine passive unautorisierte Verwendung besteht beispielsweise im Auskundschaften von möglichen Sicherheitslücken für eine später folgende Attacke.

5.5 Zusammenfassung

Dieses Kapitel thematisiert im Rahmen einer grundlegenden Bedrohungsanalyse als Teildisziplin des Security Engineerings mögliche Angriffe auf Grids, wobei vor allem eine Klassifizierung dieser im Mittelpunkt steht.

Abschnitt 5.1 ordnet eingangs die Bedrohungsanalyse in den übergeordneten Zusammenhang des Security Engineerings ein, das sich in Ermangelung spezieller Verfahren an allgemeine, methodische und in der Regel top-down orientierte Vorgehensmodelle des Software-Engineerings anlehnt. Neben der Bedrohungsanalyse werden ebenso die übrigen Schritte und Phasen dieser Disziplin kurz beschrieben.

Mit Abschnitt 5.2 schließt sich eine Diskussion wichtiger früherer Arbeiten auf dem Gebiet der Klassifikation von Angriffen an. Es fällt auf, dass die Ansätze alle relativ ähnlich sind, wobei zumeist eine Klassifizierung nach den Angriffstechniken verfolgt wird. Unterschiede ergeben sich allenfalls in der Granularität der Klassen. Da Grids prinzipiell eine Weiterentwicklung des verteilten Rechnens darstellen, sind bisherige Klassifizierungen grundsätzlich auch übertragbar.

Vor einer Untersuchung expliziter Angriffe erfasst Abschnitt 5.3 zunächst die möglichen Angreifer, wobei diese individuellen Klassen zugeordnet werden. In Abschnitt 5.4 werden die im Grid anzunehmenden Angriffe in ein adäquates, möglichst allgemeingültiges Schema eingeordnet. Zu diesem Zweck werden drei Klassen in Anlehnung an die weithin anerkannten Gefährdungsbereiche nach Eckert festgelegt, wobei vor allem auch geeignete Beispiele für Angriffe gegeben werden.

KAPITEL SECHS

Zusammenfassung und Ausblick

Inhalt des Kapitels

6.1 Zusammenfassung 83
6.2 Offene Fragestellungen 84

Zum Abschluss der Arbeit fasst Abschnitt 6.1 zunächst noch einmal die vorliegende Arbeit zusammen. Danach geht Abschnitt 6.2 auf einige offene Fragestellungen ein, die als Grundlage für zukünftige Arbeiten dienen könnten.

6.1 Zusammenfassung

Mit der größer werdenden Verbreitung von Grids muss gleichzeitig ein tragfähiges, organisationsübergreifendes Sicherheitsmanagement etabliert werden. Besonders auch im neuartigen und noch stetigem Wandel unterworfenen Umfeld von Grids sind hierfür Intrusion Detection Systeme von großer Bedeutung, wo sie idealerweise etwaige schon bestehende Sicherheitslösungen ergänzen. Mit dem weiteren Anstieg der Leistungsfähigkeit von Grids müssen diese geradezu als „magische“ Anziehungspunkte aufgefasst werden, wobei bedingt durch die charakteristische große Komplexität vielfältige Szenarien missbräuchlicher Nutzung realitätsnah erscheinen. Eine ausgereifte Einbruchserkennung ist nicht zuletzt auch vor dem Hintergrund einer weiter zunehmenden industriellen Nutzung von hohem Wert für eine dauerhafte Überlebensfähigkeit von Grids.

In dieser Diplomarbeit wurden im Hinblick auf eine zukünftige Realisierung generische Anforderungen an gridbasierte Intrusion Detection Systeme herausgearbeitet. Im Rahmen einer grundlegenden Bedrohungsanalyse wurde außerdem eine Klassifikation möglicher Angriffe auf Grids entwickelt.

Um die Aufgabenstellung angemessen zu bearbeiten, werden eingangs in Kapitel 2 die für das Verständnis der weiteren Arbeit elementaren Grundlagen erläutert. Nach einem allgemeinen Überblick zum Sicherheitsmanagement auf Basis des „OSI Management Frameworks“ werden die Eigenschaften und die verschiedenen Ausprägungen von Intrusion Detection Systemen sowie das Themenfeld des Grid Computing dargestellt, wobei auch die besondere Problemstellung bei der Betrachtung von Intrusion Detection Systemen in komplexen Grid-Umgebungen erläutert wird.

Kapitel 3 beleuchtet den aktuellen Entwicklungsstand gridbasierter Intrusion Detection Systeme. Zunächst werden hierzu zumindest die wichtigsten Konzepte der den gridspezifischen Lösungen vorausgehenden verteilten Einbruchserkennung analysiert, die verglichen mit klassischen Mechanismen einen bedeutenden Fortschritt besonders hinsichtlich des domänen- oder sogar organisationsübergreifenden Einsatzes bedeutet. Da dennoch neue Lösungen benötigt werden, werden schließlich Forschungsprojekte zu neuen gridweiten Intrusion Detection Systemen diskutiert. Es zeigt sich, dass

trotz einiger Bemühungen derzeit noch keine umfassende und praktikable Lösung vorgestellt wurde. Vielmehr konzentrieren sich die Ansätze jeweils nur auf einzelne Teilaspekte der gesamten Problematik im Grid, ohne vorher ein angemessenes Fundament und hier besonders die grundsätzlichen Anforderungen im Umfeld von Grids begründet zu haben.

In der Folge wendet sich Kapitel 4 der Analyse zur Gewinnung von wichtigen generischen Anforderungen an gridbasierte Systeme zur Einbruchserkennung zu. Damit eine Applikabilität des Verfahrens in Bezug auf potenzielle Zielumgebungen gewährleistet werden kann, werden generell hohe Ansprüche an das Vorgehen gestellt. Initial werden die beteiligten Aktoren identifiziert, wobei die Komplexität von Grids zu einer Vielzahl derer führt. Die maßgeblichen Anforderungen werden sodann aus praxisnahen Anwendungsfällen abgeleitet, die, um ein möglichst weit greifendes und vollständiges Bild zu erhalten, in mehrere Dimensionen strukturiert werden. Aus diesem Vorgehen resultiert letztlich ein umfangreicher Katalog von 48 verschiedenen Anforderungen, die dann gleichfalls jeweils beschrieben und mit einer Erläuterung ihrer Relevanz versehen werden. Mit zu den wichtigsten Anforderungen zählen demnach unter anderem die Interoperabilität sowie die Portabilität und Wiederverwendbarkeit, einheitliche Schnittstellen oder auch die Wahrung der Integrität.

Nach der substantiierten Ermittlung wichtiger generischer Anforderungen an Intrusion Detection Systeme im Grid thematisiert Kapitel 5 im Rahmen einer grundlegenden Bedrohungsanalyse noch mögliche Angriffe auf Grids, wobei eine Klassifikation derselben angestrebt wird. Vorneweg wird die Bedrohungsanalyse in den übergeordneten Zusammenhang des Security Engineerings eingeordnet, das sich in Ermangelung spezieller Verfahren an allgemeine, methodische und in der Regel top-down orientierte Vorgehensmodelle des Software-Engineerings anlehnt. Ein Vergleich früherer Klassifikation führt zu dem Schluss, dass alle Konzepte relativ ähnlich und bisherige Ansätze, gegebenenfalls in angepasster Form, ebenso auf Grids übertragbar sind. Vor einer Untersuchung expliziter Angriffe werden zunächst mögliche Angreifer erfasst. Zur Klassifikation werden schließlich drei Klassen in Anlehnung an die weithin anerkannten Gefährdungsbereiche nach Eckert festgelegt, wobei vor allem auch geeignete Beispiele für Angriffe gegeben werden.

6.2 Offene Fragestellungen

Im Zuge dieser Arbeit wurden immer wieder Anknüpfungspunkte für mögliche zukünftige Arbeiten offensichtlich, die für die weitere Fortentwicklung hin zum angestrebten einsatzfähigen gridweiten Intrusion Detection System von substantieller Bedeutung sind. Die brisanteren davon werden im folgenden kurz vorgestellt.

Die angeführten Anwendungsfälle und der sich daraus gebildete Katalog einer Vielzahl an Anforderungen zeigen eindrucksvoll, welch große Herausforderung die Schaffung einer gridbasierter Erkennung von Angriffen generell darstellt. Um dieses Ziel zu erreichen, kann es nun Bestandteil anschließender Arbeiten sein, den begonnenen Softwareentwicklungsprozess hin zum letztlich finalen, produktiv nutzbaren Produkt schrittweise weiterzuführen. Als dienlicher Zwischenschritt sollte zunächst versucht werden, eine detaillierte Architekturspezifikation auszuarbeiten, die alle relevanten Aspekte betreffend die Einbruchserkennung im Grid in sich vereint. Anhand eines wohlüberlegt ausgewählten Vorgehensmodells (siehe hierzu u.a. [RP02]) sollte dann die Implementierung, Validierung sowie die letztliche Systemeinführung wirksam vorangebracht werden können.

Für eine erfolgreiche Erkennung von Angriffen sollten genauso auch die begonnenen Anstrengungen auf dem Gebiet des Security Engineerings zu Ende gebracht werden. An die Bedrohungsanalyse sollte sich hierzu eine eingehende Bewertung der für Grids relevanten Bedrohungen im Sinne einer standardisierten Risikoanalyse anschließen. Auf Basis einer solchen Quantifizierung potenzieller

primärer oder ebenfalls sekundärer Schäden können vordringliche Sicherheitsmechanismen gezielt frühzeitig erkannt und in der Folge behandelt werden.

Der Fokus der vorliegenden Arbeit liegt ausschließlich auf der gridweiten Erkennung von Einbrüchen. Ein überaus lohnendes Betätigungsfeld speziell im Kontext der besonders sicherheitsbedürftigen Grids stellen überdies auch die auf Intrusion Detection Systemen aufsetzenden Intrusion Response Systeme (IRS) (vgl. [IRS]) dar, mit dessen Hilfe auf identifizierte Angriffe gegenüber dem ansonsten manuellen Eingreifen von zuständigen Administratoren automatisiert reagiert werden könnte. Gerade in derart großen Umgebungen, wie denjenigen der Grids, erweist es sich in punkto eines erfolgreichen Sicherheitsmanagements als äußerst hilfreich und zweckmäßig, manuelle Eingriffe konsequent so weit als möglich zu reduzieren. Auch wenn dies für herkömmliche Szenarien vernetzter Systeme bereits hinreichend erforscht wurde, sind im Bereich von Grids doch wichtige Fragestellungen offen (vgl. [RWH+07]). Etwa, welche Gegenmaßnahmen unter ausdrücklicher Beachtung einschränkender rechtlicher Rahmenbedingungen dafür geeignet erscheinen, die Sicherheit von Grids effektiv zu verbessern, und wie diese schließlich im Grid umgesetzt werden könnten. Einer realen Implementierung würde mit Interesse entgegengesehen werden.

Literaturverzeichnis

[ABK+05] Asadzadeh, Parvin, Rajkumar Buyya, Chun Ling Kei, Deepa Nayar, and Srikumar Venugopal: *Global Grids and Software Toolkits: A Study of Four Grid Middleware Technologies*, chapter 22, pages 431–458. John Wiley & Sons, June 2005, ISBN 0-471-65471-X.

[ABKM01] Andersen, David, Hari Balakrishnan, Frans Kaashoek, and Robert Morris: *Resilient overlay networks*. In *Proceedings of the 18th Symposium on Operating Systems Principles (SOSP)*, pages 131–145, May 2001.

[ACDM01] Arvidsson, J., A. Cormack, Y. Demchenko, and J. Meijer: *Incident object description and exchange format requirements*. RFC 3067, February 2001. `http://www.ietf.org/rfc/rfc3067.txt`.

[AFV95] Anderson, D., T. Frivold, and A. Valdes: *Next-generation intrusion detection expert system (nides): A summary*. Technical Report SRI-CSL-95-07, SRI International, Computer Science Laboratory, May 1995.

[AIA05] Alim, A., I. Ismail, and S. Ahmed: *Idsuda: An intrusion detection system using distributed agents*. International Journal on Computer Network and Internet Research, 5(1):1–11, December 2005.

[AID] *Aide – advanced intrusion detection environment*. `http://www.cs.tut.fi/~rammer/aide.html`.

[ALJ+95] Anderson, D., T. Lunt, H. Javitz, A. Tamaru, and A. Valdes: *Detecting unusual program behavior using the statistical component of the next-generation intrusion detection expert system (nides)*. Technical Report SRI-CSL-95-06, SRI International, Computer Science Laboratory, May 1995.

[Amo94] Amoroso, Edward: *Fundamentals of Computer Security Technology*. Prentice-Hall, 1994.

[And80] Anderson, James P.: *Computer security threat monitoring and surveillance*. Technical report, Fort Washington, PA, USA, April 1980. `http://csrc.nist.gov/publications/history/ande80.pdf`.

[AT05] Abraham, Ajith and Johnson Thomas: *Distributed Intrusion Detection Systems: A Computational Intelligence Approach*, chapter 5, pages 105–135. Idea Group Inc. Publishers, 2005.

[Axe00] Axelsson, Stefan: *Intrusion detection systems: A survey and taxonomy*. Technical Report 99-15, Chalmers University of Technology, Göteborg, Sweden, March 2000.

[BBB04] Beckles, Bruce, Sacha Brostoff, and Stuart Ballard: *A first attempt: initial steps toward determining scientific users' requirements and appropriate security paradigms for computational grids*. In *Proceedings of the Workshop on Requirements Capture for Collaboration in e-Science*, pages 17–43, 2004.

[BCR+06] Beckles, B., P. V. Coveney, P. Y. A. Ryan, A. E. Abdallah, S. M. Pickles, J. M. Brooke, and M. McKeown: *A user-friendly approach to computational grid security*. In *Proceedings of the UK e-Science All Hands Meeting*, September 2006. http://www.allhands.org.uk/2006/proceedings/papers/636.pdf.

[BDE+07] Büchner, O., C. Dohmen, T. Eifert, H. Enke, T. Fieseler, A. Frank, S. Freitag, A. Garcia, C. Grimm, W. Gürich, H. Heller, T. Jejkal, H. Nitsche, O. Schneider und W. Ziegler: *Betriebskonzept für die D-Grid Infrastruktur*. Technischer Bericht, D-Grid Integrationsprojekt (DGI), 2007.

[BF92] Borenstein, N. and N. Freed: *Mime (multipurpose internet mail extensions): Mechanisms for specifyingand describing the format of internet message bodies*. RFC 1341, June 1992. http://www.ietf.org/rfc/rfc1341.txt.

[BGFI+98] Balasubramaniyan, Jai Sundar, Jose Omar Garcia-Fernandez, David Isacoff, Eugene Spafford, and Diego Zamboni: *An architecture for intrusion detection using autonomous agents*. In *Proceedings of the 14th Annual Computer Security Applications Conference (ACSAC)*, pages 13–24, Washington, DC, USA, December 1998. IEEE Computer Society, ISBN 0-8186-8789-4.

[Bio] *Biogrid – general repository for interaction datasets*. http://www.thebiogrid.org.

[BMB] *Bundesministerium für Bildung und Forschung (BMBF)*. http://www.bmbf.de.

[BNP03] Bubak, Marian, Piotr Nowakowski, and Robert Pajak: *An overview of european grid projects*. In *Proceedings of the 1st European Across Grids Conference*, pages 299–308, February 2003.

[BOvdgFGR06] Baur, Timo, Nils O. v. d. gentschen Felde, Markus Garschhammer und Helmut Reiser: *Monitoring-Anwendungsfälle und Anforderungen*. In: Rückemann, Claus Peter (Herausgeber): *Ergebnisse der Studie und Anforderungsanalyse in den Fachgebieten Monitoring, Accounting, Billing bei den Communities und Ressourcenanbietern im D-Grid*, Kapitel 4, Seiten 21–44. Juni 2006.

[BS95] Brinkley, Donald L. and Roger R. Schell: *What is there to worry about? An introduction to the computer security problem*, pages 11–39. IEEE Computer Society Press, Los Alamitos, CA, USA, 1995.

[BSD+91] Brentano, J., S. R. Snapp, G. V. Dias, T. L. Goan, L. T. Heberlein, C. L. Ho, K. N. Levitt, B. Mukherjee, and S. E. Smaha: *An architecture for a distributed intrusion detection system*. In *Proceedings of the 14th Department of Energy Computer Security Group Conference*, pages 25–45, May 1991.

[BSK03] Bellovin, S., J. Schiller, and C. Kaufman: *Security mechanisms for the internet*. RFC 3631, December 2003. http://www.ietf.org/rfc/rfc3631.txt.

[BWB05] Beckles, B., V. Welch, and J. Basney: *Mechanisms for increasing the usability of*

grid security. International Journal of Human-Computer Studies, 63(1–2):74–101, July 2005.

[Cat92] Catlett, Charles E.: *In search of gigabit applications*. IEEE Communications Magazine, 30(4):42–51, April 1992, ISSN 0163-6804.

[CB02] Chetty, Madhu and Rajkumar Buyya: *Weaving computational grids: How analogous are they with electrical grids?* Computing in Science and Engineering, 4(4):61–71, July/August 2002, ISSN 1521-9615. `http://www.gridbus.org/papers/WeavingGrid.pdf`.

[CCD+99] Cheung, Steven, Rick Crawford, Mark Dilger, Jeremy Frank, Jim Hoagland, Karl Levitt, Jeff Rowe, Stuart Staniford-Chen, Raymond Yip, and Dan Zerkle: *The design of grids: A graph-based intrusion detection system*. Technical report, University of California at Davis, January 1999.

[CER] *Cern – the world's largest particle physics laboratory*. `http://public.web.cern.ch`.

[CNE] *Cnes (centre national d'Études spatiales)*. `http://www.cnes.fr`.

[Con] *Condor – high throughput computing (htc)*. `http://www.cs.wisc.edu/condor`.

[Con02] ConSecur: *Einführung von Intrusion-Detection-Systemen – Grundlagen*. Technischer Bericht, Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn, Oktober 2002. `http://www.bsi.de/literat/studien/ids02/dokumente/Grundlagenv10.pdf`.

[CS95] Crosbie, Mark and Gene Spafford: *Defending a computer system using autonomous agents*. In *Proceedings of the 18th National Information Systems Security Conference*, pages 549–558, October 1995.

[CS03] Choon, Ong Tian and Azman Samsudin: *Grid-based intrusion detection system*. In *Proceedings of the 9th Asia-Pacific Conference on Communications (APCC)*, volume 3, pages 1028–1032, September 2003.

[D-G] *Die Deutsche Grid-Initiative (D-Grid)*. `http://www.d-grid.de`.

[D-G04] D-Grid-Initiative: *e-Science in Deutschland: F&E-Rahmenprogramm 2005 bis 2009*. Technischer Bericht, Juli 2004. `http://grid.desy.de/d-grid/RahmenprogrammEndfassung.pdf`.

[DAT+99] Dierks, T., C. Allen, W. Treese, P. L. Karlton, A.O. Freier, and P. C. Kocher: *The tls protocol version 1.0*. RFC 2246, January 1999. `http://www.ietf.org/rfc/rfc2246.txt`.

[DCF07] Debar, H., D. Curry, and B. Feinstein: *The intrusion detection message exchange format (idmef)*. RFC 4765, March 2007. `http://www.ietf.org/rfc/rfc4765.txt`.

[Den87] Denning, Dorothy E.: *An intrusion-detection model*. IEEE Transactions on Software Engineering, 13(2):222–232, February 1987, ISSN 0098-5589.

[Dil] *Diligent – a digital library infrastructure on grid enabled technology*. `http://www.diligentproject.org`.

[DSh] *Dshield – cooperative network security community*. http://www.dshield.org.

[Eas01] Eastlake, D.: *Us secure hash algorithm 1 (sha1)*. RFC 3174, September 2001. http://www.ietf.org/rfc/rfc3174.txt.

[Eck06] Eckert, Claudia: *IT-Sicherheit. Konzepte – Verfahren – Protokolle*. Oldenbourg Wissenschaftsverlag, München, 4. Auflage, April 2006, ISBN 3-486-57851-0.

[EDG] *European datagrid – computing for data intensive science*. http://eu-datagrid.web.cern.ch.

[EGE] *Egee (enabling grids for e-science)*. http://www.eu-egee.org.

[ER02] Erler, Thomas und Michael Ricken: *UML – Das bhv Taschenbuch*. vmi Buch, Bonn, 1. Auflage, September 2002, ISBN 3-8266-8066-9.

[EUR] *Eurogrid – application testbed for european grid computing*. http://www.eurogrid.org.

[FB96a] Freed, N. and N. Borenstein: *Multipurpose internet mail extensions (mime) part one: Format of internetmessage bodies*. RFC 2045, November 1996. http://www.ietf.org/rfc/rfc2045.txt.

[FB96b] Freed, N. and N. Borenstein: *Multipurpose internet mail extensions (mime) part two: Media types*. RFC 2046, November 1996. http://www.ietf.org/rfc/rfc2046.txt.

[FDL+06] Feng, Guofu, Xiaoshe Dong, Weizhe Liu, Ying Chu, and Junyang Li: *Ghids: Defending computational grids against misusing of shared resources*. In *Proceedings of the IEEE Asia-Pacific Conference on Services Computing (APSCC)*, pages 526–533, 2006.

[FK98] Foster, Ian and Carl Kesselman (editors): *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 1st edition, 1998, ISBN 1-55860-475-8.

[FK04] Foster, Ian and Carl Kesselman (editors): *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 2nd edition, 2004, ISBN 1-55860-933-4.

[FKT01] Foster, Ian, Carl Kesselman, and Steven Tuecke: *The anatomy of the grid: Enabling scalable virtual organizations*. International Journal of Supercomputer Applications, 15(3), 2001. http://www.globus.org/alliance/publications/papers/anatomy.pdf.

[FM07] Feinstein, B. and G. Matthews: *The intrusion detection exchange protocol (idxp)*. RFC 4767, March 2007. http://www.ietf.org/rfc/rfc4767.txt.

[Fos02a] Foster, Ian: *The grid: A new infrastructure for 21st century science*. Physics Today, 55(2):42–47, February 2002.

[Fos02b] Foster, Ian: *What is the grid? a three point checklist*. GRIDtoday, 1(6), July 2002. http://www-fp.mcs.anl.gov/~foster/Articles/WhatIsTheGrid.pdf.

[FS06] Filipovic, Bartol und Tobias Straub: *Grid Security Infrastructure – ein Überblick*. In: *DFN Tagungsband 2006*, Seiten 115–126. Fraunhofer-Institut für Sichere Informationstechnologie, Darmstadt, Juni 2006.

[GBF+01] Gupta, D., T. C. Buchheim, B. S. Feinstein, G. A. Matthews, and R. A. Pollock: *Iap: Intrusion alert protocol*. IETF Internet-Draft, February 2001. `http://www3.ietf.org/proceedings/01mar/I-D/idwg-iap-04.txt`.

[GE06] Graaff, A. J. and A. P. Engelbrecht: *Optimized coverage of non-self with evolved lymphocytes in an artificial immune system*. International Journal of Computational Intelligence Research, 2(2):127–150, 2006.

[GHN+04] Graham, Paul, Matti Heikkurinen, Jarek Nabrzyski, Ariel Oleksiak, Mark Parsons, Heinz Stockinger, Kurt Stockinger, Maciej Stroiski, and Jan Wglarz: *Eu funded grid development in europe*. In *Proceedings of the 2nd European Across Grids Conference*, 2004.

[gLi] *glite – lightweight middleware for grid computing*. `http://glite.web.cern.ch`.

[Glo] *Globus toolkit – open source software toolkit used for building grid systems and applications*. `http://www.globus.org`.

[GORP04] Gerloni, Helmar, Barbara Oberhaitzinger, Helmut Reiser und Jürgen Plate: *Praxisbuch – Sicherheit für Linux-Server und -Netze*. Carl Hanser Verlag, München, Wien, 2004, ISBN 3-446-22626-5. `http://www.nm.ifi.lmu.de/~sicherheitsbuch`.

[GRI] *Grid Interoperability Project (GRIP)*. `http://www.dwd.de/de/Technik/Projekte/GRIP`.

[HAN99] Hegering, Heinz Gerd, Sebastian Abeck, and Bernhard Neumair: *Integrated Management of Networked Systems*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 1999, ISBN 3-932588-16-9.

[HFAAA05] Hegazy, I. M., H. M. Faheem, T. Al-Arif, and T. Ahmed: *Evaluating how well agent-based ids perform*. IEEE Potentials, 24(2):27–30, 2005.

[HML92] Heberlein, L. Todd, Biswanath Mukherjee, and Karl N. Levitt: *Internet security monitor: An intrusion-detection system for large-scale networks*. In *Proceedings of the 15th National Computer Security Conference*, pages 262–271, October 1992.

[IRS] *Intrusion Detection Systeme vs. Intrusion Response Systeme*. `http://einstein.informatik.uni-oldenburg.de/23653.html`.

[JS99] Jones, Anita K. and Robert S. Sielken: *Computer system intrusion detection: A survey*. Technical report, University of Virginia, 1999. `http://www.cs.virginia.edu/~jones/IDS-research/Documents/jones-sielken-survey-v11.pdf`.

[Ken99] Kendall, K.: *A database of computer attacks for the evaluation of intrusion detection systems*. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusets Institute of Technology (MIT), Cambridge, MA, USA, June 1999.

[KLL+97] Karger, David, Eric Lehman, Tom Leighton, Matthew Levine, Daniel Lewin, and Rina Panigrahy: *Consistent hashing and random trees: Distributed caching proto-*

cols for relieving hot spots on the world wide web. In *Proceedings of the 29th ACM Symposium on Theory of Computing (STOC)*, pages 654–663, May 1997.

[LA04] Laszewski, Gregor von and Kaizar Amin: *Grid Middleware*, chapter 5, pages 109–130. John Wiley & Sons, June 2004, ISBN 0-470-86206-8. `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski-gridmiddleware.pdf`.

[Lac74] Lackey, R. D.: *Penetration of computer systems. an overview*. Honeywell Computer Journal, 8(2):81–85, 1974.

[Lan85] Landreth, B.: *Out of the Inner Circle, A Hacker's Guide to Computer Security*. Microsoft Press, Bellevue, WA, USA, 1985.

[LCG] *Lcg (lhc computing grid) – distributed production environment for physics data processing*. `http://lcg.web.cern.ch/LCG`.

[LHC] *Lhc (large hadron collider)*. `http://lhc.web.cern.ch/lhc`.

[Li05] Li, Tao: *An immunity based network security risk estimation*. Science in China Series F, 48(5):557–578, 2005.

[LJ97] Lindqvist, Ulf and Erland Jonsson: *How to systematically classify computer security intrusions*. In *Proceedings of the IEEE Symposium on Security and Privacy*, page 154, Washington, DC, USA, 1997. IEEE Computer Society.

[LLLY05a] Leu, Fang Yie, Ming Chang Li, Jia Chun Lin, and Fu Yi Yang: *Integrating grid with intrusion detection*. In *Proceedings of the 19th IEEE International Conference on Advanced Information Networking and Applications (AINA)*, pages 304–309, March 2005.

[LLLY05b] Leu, Fang Yie, Jia Chun Lin, Ming Chang Li, and Chao Tung Yang: *A performance-based grid intrusion detection system*. In *Proceedings of the 29th International Computer Software and Applications Conference (COMPSAC)*, volume 1, pages 525–530, July 2005, ISBN 0-7695-2413-3.

[LOR+03] Leite, R., F. Oliveira, C. Ribeiro, J. Oliveira, B. Schulze, and E. Madeira: *Functionalities in grid computing with active services*. In *Proceedings of the 1st International Workshop on Middleware for Grid Computing*, pages 194–199, June 2003.

[LPO+03] Liabotis, I., O. Prnjat, T. Olukemi, A. Ching, A. Lazarevic, L. Sacks, M. Fisher, and P. McKee: *Self-organising management of grid environments*. In *Proceedings of the International Symposium on Telecommunications (IST)*, 2003.

[LRZ] *Leibniz-Rechenzentrum München (LRZ)*. `http://www.lrz.de`.

[MBA02] Murshed, M., R. Buyya, and D. Abramson: *Gridsim: A grid simulation toolkit for resource management and scheduling in large-scale grid computing environments*. In *Proceedings of the 17th IEEE International Symposium on Parallel and Distributed Processing (IPDPS)*, April 2002.

[MDD03] Meijer, J., R. Danyliw, and Y. Demchenko: *The incident data exchange format data model and xml implementation documenttype definition*. IETF Internet-Draft, March 2003. `http://tools.ietf.org/id/draft-ietf-inch-iodef-01.txt`.

[MHL94] Mukherjee, Biswanath, L. Todd Heberlein, and Karl N. Levitt: *Network intrusion detection*. IEEE Network, 8(3):26–41, May/June 1994.

[MVB03] Morakis, Evangelos, Stilianos Vidalis, and Andrew Blyth: *A framework for representing and analysing cyber attacks using object oriented hierarchy trees*. In *Proceedings of the 2nd European Conference On Information Warfare And Security (ECIW)*, July 2003.

[NAR] *Naregi (national research grid initiative)*. `http://www.naregi.org`.

[NLSX07] Ni, Jiancheng, Zhishu Li, Jirong Sun, and Jianchuan Xing: *Self-adaptive intrusion detection system for computational grid*. In *Proceedings of the 1st IEEE & IFIP International Symposium on Theoretical Aspects of Software Engineering (TASE)*, pages 97–106, June 2007.

[NP89] Neumann, Peter G. and Donn B. Parker: *A summary of computer misuse techniques*. In *Proceedings of the 12th National Computer Security Conference*. National Institute of Standards and Technology / National Computer Security Center, October 1989.

[NR05] Naqvi, Syed and Michel Riguidel: *Threat model for grid security services*. In *Proceedings of the European Grid Conference*, pages 1048–1055, Berlin / Heidelberg, February 2005. Springer-Verlag, ISBN 978-3-540-26918-2.

[NYHR05] Neuman, C., T. Yu, S. Hartman, and K. Raeburn: *The kerberos network authentication service (v5)*. RFC 4120, July 2005. `http://www.ietf.org/rfc/rfc4120.txt`.

[Opea] *Opendht – a publicly accessible dht service*. `http://www.opendht.org`.

[Opeb] *Openmolgrid – open computing grid for molecular science and engineering*. `http://www.openmolgrid.org`.

[OvdgF05] Felde, Nils O. v. d. gentschen: *Leistungsfähigkeit von Anomalieerkennungsverfahren in domänenübergreifenden Meta-IDS*. Diplomarbeit, Rheinische Friedrich-Wilhelms-Universität, Bonn, März 2005.

[OvdgFHS06] Felde, Nils O. v. d. gentschen, Heinz Gerd Hegering, and Michael Schiffers: *IT Service Management Across Organizational Boundaries*, chapter 3, pages 147–175. Springer-Verlag, Berlin / Heidelberg, 2006, ISBN 3-540-34128-5.

[PN97] Porras, Phillip A. and Peter G. Neumann: *Emerald: Event monitoring enabling responses to anomalous live disturbances*. In *Proceedings of the 20th National Information Systems Security Conference*, pages 353–365, October 1997.

[Pre] *Prelude – the hybrid ids framework*. `http://www.prelude-ids.org`.

[Ric03] Richardson, R.: *Csi/fbi: Computer crime and security survey*. Technical report, CSI, Computer Security Institue, 2003.

[Riv92] Rivest, R.: *The md5 message-digest algorithm*. RFC 1321, April 1992. `http://www.ietf.org/rfc/rfc1321.txt`.

[RM05] Riedel, Morris and Daniel Mallmann: *Standardization processes of the unicore grid system*. In *Proceedings of the 1st Austrian Grid Symposium*, pages 191–203, 2005.

[Ros01] Rose, M.: *The blocks extensible exchange protocol core*. RFC 3080, March 2001. `http://www.ietf.org/rfc/rfc3080.txt`.

[RP02] Rechenberg, Peter und Gustav Pomberger (Herausgeber): *Informatik-Handbuch*. Carl Hanser Verlag, München / Wien, 3. Auflage, 2002, ISBN 3-446-21842-4.

[RWH+07] Rowanhill, Jonathan, Glenn Wasson, Jim Hill, Zach Basney, Yuliyan Kiryakov, John Knight, Anh Nguyen-Tuong, Andrew Grimshaw, and Marty Humphrey: *Dynamic system-wide reconfiguration of grid deployments in response to intrusion detections*. In *Proceedings of the 3rd High Performance Computing Conference (HPCC)*, September 2007.

[Sam] *Samhain – open source solution for host-based intrusion detection*. `http://www.la-samhna.de/samhain`.

[SBD+91a] Snapp, Steven R., James Brentano, Gihan V. Dias, Terrance L. Goan, Tim Grance, L. Todd Heberlein, Che Lin Ho, Karl N. Levitt, Biswanath Mukherjee, Douglass L. Mansur, Kenneth L. Pon, and Stephen E. Smaha: *A system for distributed intrusion detection*. In *Proceedings of the IEEE COMPCON*, pages 170–176, February/-March 1991.

[SBD+91b] Snapp, Steven R., James Brentano, Gihan V. Dias, Terrance L. Goan, L. Todd Heberlein, Che Lin Ho, Karl N. Levitt, Biswanath Mukherjee, Stephen E. Smaha, Tim Grance, Daniel M. Teal, and Doug Mansur: *Dids (distributed intrusion detection system) – motivation, architecture, and an early prototype*. In *Proceedings of the 14th National Computer Security Conference*, pages 167–176, October 1991.

[SCCC+96] Staniford-Chen, S., S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. Levitt, C. Wee, R. Yip, and D. Zerkle: *Grids – a graph-based intrusion detection system for large networks*. In *Proceedings of the 19th National Information Systems Security Conference*, 1996.

[Sch00] Schneier, Bruce: *Secrets and Lies: Digital Security in a Networked World*. John Wiley & Sons, 2000, ISBN 0-4712-5311-1.

[SEL+05] Streit, A., D. Erwin, Th. Lippert, D. Mallmann, R. Menday, M. Rambadt, M. Riedel, M. Romberg, B. Schuller, and Ph. Wieder: *Unicore – from project results to production grids*. Technical report, John von Neumann-Institue for Computing, Jülich, Germany, 2005.

[SMK+01] Stoica, Ion, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan: *Chord: A scalable peer-to-peer lookup service for internet applications*. In *Proceedings of the ACM SIGCOMM Conference*, pages 149–160, 2001.

[SNKW06] Schulter, Alexandre, Fabio Navarro, Fernando Koch, and Carlos Becker Westphall: *Towards grid-based intrusion detection*. In *Proceedings of the 10th Network Operations and Management Symposium (NOMS)*, pages 1–4, April 2006, ISBN 1-4244-0142-9.

[Sno] *Snort – the open source intrusion detection system*. `http://www.snort.org`.

[SNV04] Sardinha, Luis, Nuno Ferreira Neves, and Paulo Verissimo: *Tolerating intrusions in grid systems*. In *Proceedings of the International Conference on Security and Management (SAM)*, pages 207–220. CSREA Press, 2004, ISBN 1-932415-37-8. `http://www.di.fc.ul.pt/~nuno/PAPERS/SAM04.pdf`.

[Sob00] Sobirey, Michael: *Intrusion detection systems bibliography*, June 2000. `http://www.cse.sc.edu/research/isl/mirrorSobireys.shtml`.

[SRI] *Sri international – independent research, technology development and consulting organization*. http://www.sri.com.

[SRKW06] Schulter, Alexandre, Julio Albuquerque Reis, Fernando Koch, and Carlos Becker Westphall: *A grid-based intrusion detection system*. In *Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL)*, April 2006.

[SS02] Sherlia Shi, Yunxi: *Design of Overlay Networks for Internet Multicast*. Dissertation, Washington University, Saint Louis, MO, USA, August 2002.

[SSGT92] Snapp, S. R., S. E. Smaha, T. Grance, and D. M. Teal: *The dids (distributed intrusion detection system) prototype*. In *Proceedings of the USENIX Technical Conference*, pages 227–233, June 1992.

[SWDC97] Stevens, Rick, Paul Woodward, Tom DeFanti, and Charles E. Catlett: *From the i-way to the national technology grid*. Communications of the ACM, 40(11):50–60, November 1997, ISSN 0001-0782.

[SZ00] Spafford, Eugene and Diego Zamboni: *Intrusion detection using autonomous agents*. Computer Networks, 34(4):547–570, October 2000.

[TAWTAS05a] Tolba, Mohamed F., Mohammad S. Abdel-Wahab, Ismail A. Taha, and A. M. Al-Shishtawy: *Distributed intrusion detection system for computational grids*. In *Proceedings of the 2nd International Conference on Intelligent Computing and Information Systems*, March 2005.

[TAWTAS05b] Tolba, Mohamed F., Mohammad S. Abdel-Wahab, Ismail A. Taha, and A. M. Al-Shishtawy: *Gida: Toward enabling grid intrusion detection systems*. In *Proceedings of the 5th IEEE International Symposium on Cluster Computing and the Grid*, May 2005.

[TBK03] Talwar, Vanish, Sujoy Basu, and Raj Kumar: *An environment for enabling interactive grids*. In *Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing (HPDC)*, pages 184–195, June 2003.

[TCP81] *Transmission control protocol*. RFC 793, September 1981. http://www.ietf.org/rfc/rfc793.txt.

[Tri] *Tripwire*. http://www.tripwire.org.

[Tue01] Tuecke, S.: *Grid security infrastructure (gsi) roadmap*. IETF Internet-Draft, February 2001. http://www.ggf1.nl/abstracts/SEC/draft-ggf-gsi-roadmap-02.pdf.

[UNI] *Unicore (uniform interface to computing resources)*. http://www.unicore.eu.

[YBJ04] Yegneswaran, Vinod, Paul Barford, and Somesh Jha: *Global intrusion detection in the domino overlay system*. In *Proceedings of the 11th Network and Distributed System Security Symposium (NDSS)*, 2004.

[ZKL05] Zhou, Chenfeng V., Shanika Karunasekera, and Christopher Leckie: *A peer-to-peer collaborative intrusion detection system*. In *Proceedings of the IEEE International Conference on Networks (ICON)*, pages 118–123, November 2005.

[ZKL07] Zhou, Chenfeng V., Shanika Karunasekera, and Christopher Leckie: *Evaluation of a decentralized architecture for large scale collaborative intrusion detection*. In *Proceedings of the 10th IEEE International Symposium on Integrated Network Management (IM)*, pages 80–89, May 2007. http://www.csse.unimelb.edu.au/~cvzhou/pub/im07.pdf.