Ein Grid-basiertes, föderiertes Intrusion Detection System zur Sicherung der D-Grid Infrastruktur (GIDS)

Architekturkonzept für ein Grid-basiertes IDS (MS 16-1)
Meilenstein zum Abschluss des Arbeitspakets 5

Autoren:
Dr. Wolfgang Hommel (Leibniz-Rechenzentrum)
Dr. Nils gentschen Felde (Ludwig-Maximilians-Universität München)
Felix von Eye (Leibniz-Rechenzentrum)
Jan Kohlrausch (DFN-CERT GmbH)
Christian Szongott (Regionales Rechenzentrum für Niedersachsen)

GEFÖRDERT VOM

Inhaltsverzeichnis

Kapitel 1

Einleitung

Dieses Dokument präsentiert als zusammenfassendes Ergebnis des Arbeitspakets 5 des Projekts „Ein Grid-basiertes, föderiertes Intrusion Detection System zur Sicherung der D-Grid Infrastruktur“ (GIDS) ein Feinkonzept für die Architektur für ein Intrusion Detection Systemen (IDS) für Grids. GIDS (`http://www.grid-ids.de`) ist ein Teilprojekt im Rahmen des D-Grid (`http://www.d-grid.de`) und wird vom Bundesministerium für Bildung und Forschung (BMBF, `http://www.bmbf.de`) gefördert. Weitere Projektinformationen und Unterlagen können der Projekt-Webseite entnommen werden.

1.1 Problemstellung

Im Umfeld von Grids ergeben sich im Vergleich zu konventionellen vernetzten Systemen eine Reihe bisher ungelöster Probleme, die es im Fall des D-Grid zu bewältigen gilt. So begegnet man im Grid-Kontext unter anderem einem sehr dynamischen Umfeld. Dieses ist unter verschiedenen Gesichtspunkten festzustellen, wie zum Beispiel an einer hohen Dynamik an verfügbaren Ressourcen oder auch an hoch dynamischen Nutzergruppen beziehungsweise Virtuellen Organisationen (VO). Dies erfordert individuelle, dynamische Nutzersichten, die sich in den Kontext einer VO einbetten und deren individuellen Bedürfnissen nachkommen. Weiter ergibt sich ein Grid-typisch heterogenes Umfeld. Auch dies existiert auf mehreren Ebenen und ist unter anderem auch im Bereich der Ressourcen, der eingesetzten Grid-Middleware oder auch bei den eingesetzten Grid-Diensten zu beobachten. Nicht zuletzt die zum Teil bereits von den beteiligten Organisationen eingesetzten Sicherheitskomponenten und -werkzeuge zur Erkennung von Angriffen sind von unterschiedlichster Art.

Hier ist häufig keine Koppelung bestehender Komponenten möglich und der Grid-weite Austausch von Informationen bezüglich sicherheitsrelevanter Ereignisse wird nicht umgesetzt. Dies ist nicht nur auf die Heterogenität in diesem Umfeld zurückzuführen, sondern auch auf Randbedingungen wie beispielsweise unterschiedliche Sicherheits- und Informationsverbreitungsrichtlinien („security and information sharing policies“) der beteiligten realen Organisationen. Darüber hinaus bieten Firewalls derzeit keinen umfassenden Schutz für Grids. Aufgrund fehlender Mechanismen zur dynamischen Erkennung und Freischaltung von Kommunikationsanforderungen müssen große Portbereiche zum Teil sogar ohne einschränkende Angabe von IP-Adressen permanent freigegeben werden.

Zurzeit existiert kein Gesamtkonzept für ein kooperatives, Grid-weit föderiertes Intrusion Detection System (GIDS) mit entsprechenden Reporting-Komponenten, das sich in ein Umfeld wie dem D-Grid einbettet. Daher soll ein Konzept für ein GIDS entwickelt, im D-Grid implementiert und in die Produktion überführt werden.

1.2 Ziel

Ziel dieses Projekts ist die Bereitstellung eines GIDS-Dienstes für das D-Grid. Hierbei gilt es, soweit wie möglich bestehende Ansätze zu integrieren und ein domänen- und organisa-

tionsübergreifendes Gesamtsystem zu entwickeln. Insbesondere die Fähigkeit, mit Virtuellen Organisationen (VO) umzugehen und diese auch als Kunden in Betracht zu ziehen, ist dabei von entscheidender Bedeutung. Die Grundidee ist es, Angriffe durch die kooperative Nutzung und Auswertung von lokalen Sicherheitssystemen zu erkennen. Dazu ist der Austausch von Angriffsdaten und somit deren datenschutzkonforme Aufarbeitung, auch zur Wahrung individuell bestehender Sicherheits- und Informationsverbreitungsrichtlinien, notwendig. In einem kooperativen IDS besteht die Möglichkeit, Angriffe schneller zu erkennen, als dies mit unabhängigen und nur die lokale Sicht berücksichtigenden Sicherheitssystemen möglich ist. Somit kann eine Verkürzung der Reaktionszeit der beteiligten Parteien erzielt werden. Weiter können Vorwarnungen, an zum Zeitpunkt der Erkennung eines Angriffs noch nicht betroffenen Parteien, herausgegeben sowie gegebenfalls präventive Gegenmaßnahmen ergriffen werden.

Eine Auswertung der Daten kann sich zu großen Teilen auf bereits vorhandene Ansätze klassischer IDS stützen. Bei der Auswertung der verfügbaren Datengrundlage ist darauf zu achten, dass VO-spezifische Zugriffsrechte und Befugnisse eingehalten werden. Nach erfolgreicher Auswertung aller verfügbaren Informationen durch ein kooperatives und föderiertes GIDS, unter Beachtung individueller Sicherheits- und Datenschutz-Policies, erfolgt eine Berichterstattung über die erkannten Angriffe auf das Grid oder einzelne beteiligte Partner. Auch hier ist es von Bedeutung, dass eine VO-spezifische Sicht auf die bereitgestellten Informationen realisiert wird. Dazu ist eine Anbindung an die im D-Grid bestehenden VO Managementsysteme zu schaffen. Nach der Entwicklung einer geeigneten Architektur für ein kooperatives und föderiertes IDS in Grid-Umgebungen steht die Implementierung und Produktivführung des Systems. Es soll nach Abschluss der Projektlaufzeit ein produktives Intrusion Detection System als Grid-Dienst im D-Grid zu Verfügung stehen, das sowohl von Ressourcenanbietern als auch von Kunden (VOs, Communities etc.) genutzt werden kann.

1.3 Struktur des Dokuments

Im bisherigen Projektverlauf wurde bereits eine Grobskizze der Architektur für ein GIDS im Meilenstein 10 [8] vorgestellt. Da die in diesem Dokument vorgestellte Architektur auf dieser Grobskizze basiert, werden in Kapitel 2 die wichtigsten Aspekte des Meilensteindokuments zusammengefasst.

Kapitel 3 befasst sich dann detailliert mit den Komponenten auf Seiten der Ressourcenanbieter, während in Kapitel 4 die Komponenten auf Seiten des GIDS-Betreibers näher betrachtet werden.

Kapitel 2

Zusammenfassung Grobskizze

Nachdem Meilenstein 10 [8] bereits eine Grobskizze der Architektur von GIDS beschreibt, gibt das vorliegende Dokument einen detaillierten Überblick über den Aufbau von GIDS. Nach einer Anforderungsanalyse im Rahmen des Projekts, die in Meilenstein 6 [16] niedergeschrieben ist, ergeben sich eine Reihe an Anforderungen, die den Entwurf des GIDS maßgeblich beeinflussen. Im Folgenden beschreibt dieser Abschnitt einzelne, organisatorisch unabhängige Architekturteile, die in ihrer Summe das Grobkonzept des Grid-basierten IDS bilden. Die einzelnen dafür notwendigen Komponenten lassen sich aus den oben genannten erhobenen Anforderungen an ein GIDS ableiten, was nachfolgend in umgekehrter Reihenfolge anhand der fünf Anforderungskategorien kurz erörtert wird. Eine detaillierte Beschreibung der einzelnen Komponenten und deren Zusammenspiel ist dann in den Unterabschnitten des Kapitels 3 und des Kapitels 4 zu finden. Abbildung 2.1 stellt nochmal eine grobgranulare Übersicht der Architektur des GIDS graphisch dar.

Erkennungsleistung. Die Zweiteilung der Anforderungen zur Erkennungsleistung kann sich ebenfalls in zwei Komponenten des Systems widerspiegeln. Örtliche Aspekte können dabei durch die Einführung von *Agent*en, die verschiedene Informationsquellen (Firewalls und ihre Log-Dateien, lokale IDS-Instanzen etc.) anbinden, befriedigt werden. Die Erkennung verschiedener Angriffsmuster und -typen fordert die Notwendigkeit nach austauschbaren Analysefunktionen in einer *lokalen (G)IDS-Instanz* mit angeschlossener *GIDS-Datenbank*, um auch zeitlich lang andauernde Angriffe geeignet erkennen zu können.

Organisatorische und Datenschutzanforderungen. Zur (technischen) Durchsetzung von Informationsverbreitungsrichtlinien und der Gewährleistung von Datenschutzrichtlinien am GIDS partizipierender Ressourcenanbieter wird zum einen ein *Filter* und zum anderen ein *Anonymisierer/Pseudonymisierer* notwendig. Weiterhin erfordert die Anforderung „Weitergehende Kooperation“ eine Möglichkeit, Angriffe, für deren Entdeckung Sensoren ungeeignet sind, zu melden. Dafür bietet es sich an, einen Betreiber des GIDS einzuführen, der die notwendigen Schnittstellen bereit hält.

Sicherheitsanforderungen. Zur Realisierung kryptographischer Anforderungen bei der Interkommunikation der beteiligten Parteien bedarf es eines *GIDS-Agent*en, während die Anforderungen an eine Nutzerverwaltung und entsprechende AA-Mechanismen eine Anbindung an die bestehenden VO-Managementsysteme notwendig machen.

Nichtfunktionale Anforderungen. Aus dem umfangreichen Bereich der nichtfunktionalen Anforderungen lassen sich eine Menge Komponenten und Schnittstellen ableiten. Insbesondere folgt aus den Grid-bedingten Anforderungen ein weiteres Mal die Anbindung des GIDS an die bestehenden VO-Managementsysteme sowie auch an Monitoring-Komponenten bzw. deren Teilfunktionalität der bijektiven Abbildung von VOs zu den von ihnen im Grid genutzten Ressourcen. Durch Anforderungen wie Wiederverwendbarkeit, Erweiterbarkeit oder auch Flexibilität lässt sich einmal mehr die bereits zuvor erwähnte Komponente des *Agent*en ableiten. Zusätzlich bedingt die Forderung nach

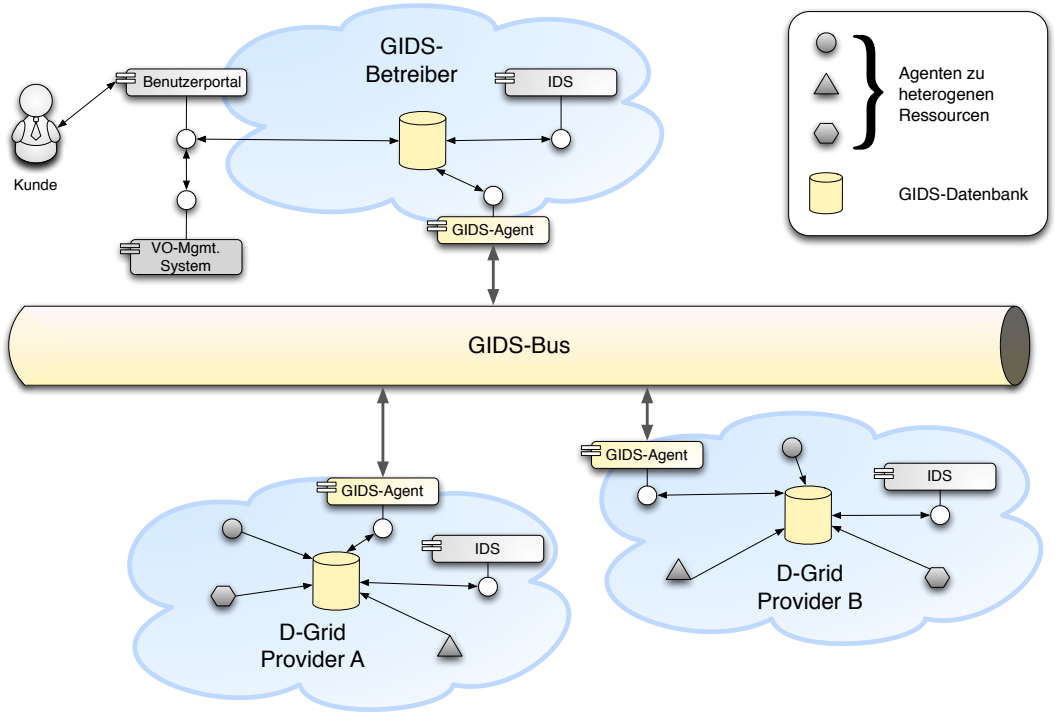

Abbildung 2.1: Grobgranulare Übersicht der Architektur des GIDS

einer großen Leistungsfähigkeit und Skalierbarkeit auch die Möglichkeit der Informationsverdichtung. Daraus folgt direkt die Notwendigkeit einer *Aggregator/Verdichter*-Komponente. Weiterhin motivieren die nichtfunktionalen Anforderungen ein einheitliches Datenaustauschformat.

Funktionale Anforderungen. Insbesondere die Einführung eines Kundenbegriffs im Rahmen Grid-basierter IDS motiviert die Existenz eines Betreibers des GIDS sowie die dazu gehörigen Komponenten. Natürlich muss auch mindestens eine *globale GIDS-Instanz* mit dazugehöriger *GIDS-Datenbank* und einem *GIDS-Agent*en zur Kommunikation gegeben sein. Zusätzlich bedarf es jedoch eines *Benutzerportal*s und einer Komponente für eine *proaktive Benachrichtigung* und deren Anbindung an VO-Managementsysteme sowie die Notwendigkeit der Abbildung von VOs zu den von ihnen genutzten Ressourcen.

Im weiteren Verlauf dieses Dokuments wird auf die genaue Funktionalität der geforderten Komponenten und deren Zusammenspiel sowie auf erste Hinweise auf eine mögliche technische Umsetzung eingegangen. Abschnitt 3.1 geht genauer auf die Architektur auf Seiten eines Ressourcenanbieters ein und Abschnitt 4.1 stellt den Aufbau auf Seiten des Betreibers des GIDS dar. Die Möglichkeit der Erweiterung des Gesamtsystems um zusätzliche Informationsanbieter ist hingegen in Meilenstein 10 [8] auf den Seiten 13 bis 15 nachzulesen.

Kapitel 3

Detaillierung der Architektur auf Seiten der Ressourcenanbieter

3.1 Architektur auf Seiten eines Ressourcenanbieters

Die grobe Übersicht über ein Grid-weites Frühwarnsystem in Abbildung 2.1 gibt bereits einen gewissen Eindruck über den Aufbau des Systems auf Seiten eines Ressourcenanbieters. Der genaue Aufbau des Systems innerhalb der administrativen Grenzen eines Ressourcenanbieters ist in Abbildung 3.1 dargestellt.

Die nachfolgend verwendeten Komponenten zum Aufbau des GIDS auf Seiten eines Ressourcenanbieters lassen sich vor allem aus den in Kapitel 3.1 des Meileinsteins 10 [8] zusammengefassten Kriterienkatalogs für IDS im Grid-Umfeld ableiten. So wird der Forderung nach der Möglichkeit der Aggregatbildung auch aus Gründen der Performanz, der Beachtung von Datenschutzaspekten (inkl. der Archivierung von Sensordaten und Berichten) und der Durchsetzung von Informationsverbreitungsrichtlinien jeweils durch die nachfolgend beschriebenen Komponenten der Datenbank, des Filters, des Aggregators bzw. Verdichters sowie des Anonymisierers und Pseudonymisierers Rechnung getragen. Die Art und Weise des Zusammenspiels dieser Komponenten resultiert insbesondere aus Grid-bedingten Anforderungen wie z.B. Dynamikaspekten und Gesichtpunkten der Leistungsfähigkeit und Performanz, während geeignete Implementierungstechniken weitere Anforderungen, wie zum Beispiel kryptographische oder die Erkennungsleistung des IDS betreffende Anforderungen, befriedigen können.

Um für ein Grid-weites IDS eine vollständige Datenreplikation gewährleisten zu können, kommt bei jedem Ressourcenanbieter eine zentrale Datenbank zum Einsatz. Diese Datenbank wird primär aus drei verschiedenen Informationsquellen gespeist:

Agent. Bei jedem Ressourcenanbieter können mehrere Agenten in verschiedenen Ausprägungen installiert sein und betrieben werden. Agenten dienen dazu, dass Informationen bestehender Sicherheitsvorkehrungen (z.B. Netflow Traces oder Firewall Logs) im sitelokalen Datenspeicher abgelegt werden können.

GIDS-Agent. In Abgrenzung zu den Agenten ist der GIDS-Agent für die Kommunikation mit den anderen Teilnehmern des GIDS verantwortlich. Zum einen verschickt er ausgewählte Informationen (siehe hierzu die Vorverarbeitungsschritte weiter unten) an andere am GIDS teilnehmende GIDS-Agenten, zum anderen empfängt er eben solche Daten von anderen GIDS-Agenten und hinterlegt sie ebenfalls in der zentralen Datenbank.

Lokale (G)IDS-Instanz. Dadurch, dass ein (lokaler) Datenbestand sämtlicher im GIDS „öffentlich“ verfügbarer Informationen vorliegt, besteht die Möglichkeit, bei jedem Ressourcenanbieter eine eigene Instanz des GIDS zu betreiben. Dadurch bedingt, dass der Site-spezifische Datenbestand unter anderem auch nicht im GIDS veröffentlichte Informationen enthalten kann, eignet sich diese Instanz des GIDS ebenfalls als mögliche Instanz eines lokalen, Site-spezifischen IDS. Berichte dieses (G)IDS werden ebenfalls in der lokalen Datenbank abgelegt.

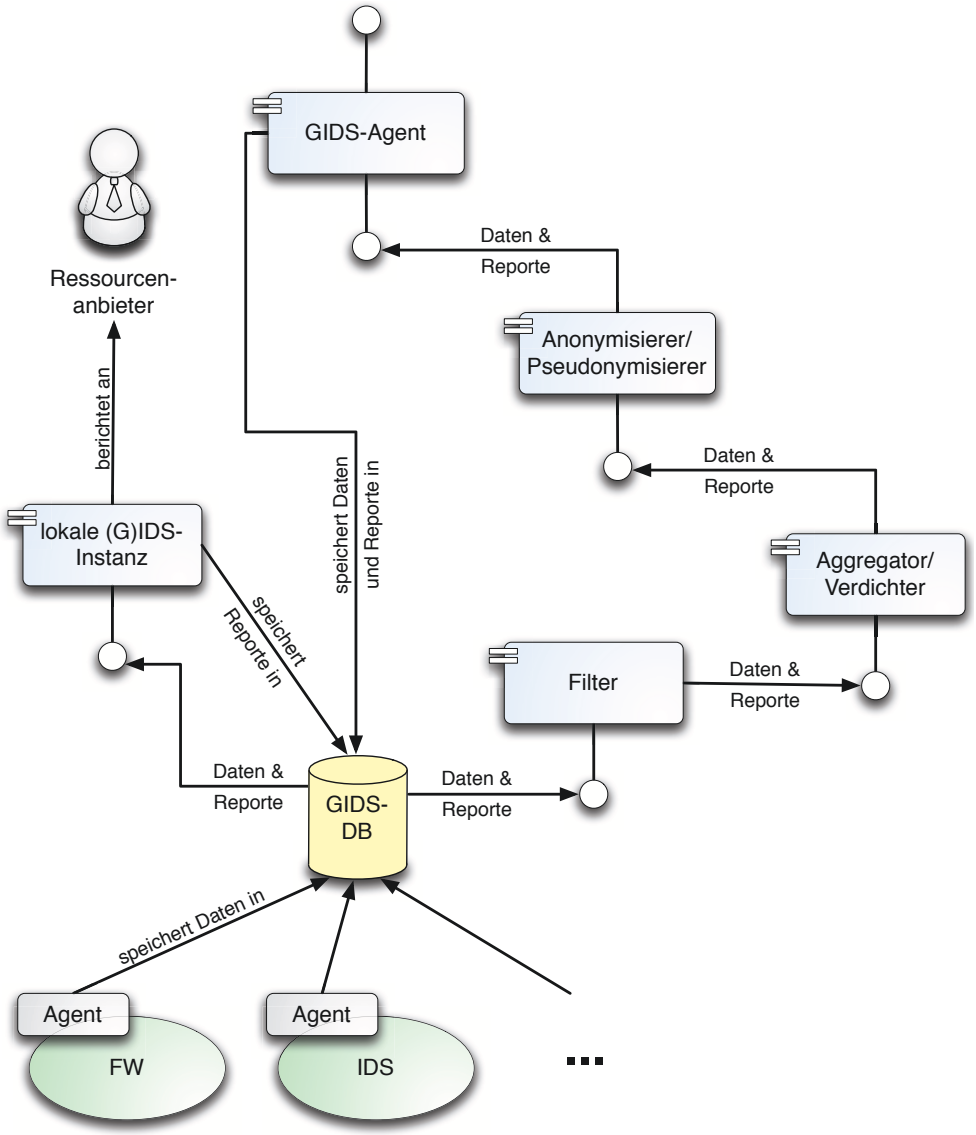

Abbildung 3.1: Architektur auf Seiten einer teilnehmenden Domäne

Bevor es zur Veröffentlichung jedweder Information, die an lokal bei einem Ressourcenanbieter gewonnen wird, im Grid durch den GIDS-Agenten kommt, durchlaufen sämtliche Informationen noch drei Vorverarbeitungsschritte.

Filter. Neue Datensätze, die in die zentrale Datenbank geschrieben werden und somit zur Weitergabe bzw. Veröffentlichung im GIDS potentiell vorgesehen sind, werden an einen Filter weitergereicht. Die primäre Aufgabe des Filters ist nun das Durchsetzen der Site-spezifischen Informationsverbreitungsrichtlinien. In Abgrenzung zu Datenschutzbestimmungen sind Informationsverbreitungsrichtlinien zumeist Bestandteil lokaler Sicherheitsrichtlinien, die zum Beispiel die Vermeidung der Verbreitung interner Topologiemerkmale, Sicherheitsverletzungen etc. fordern. Der Filter kann einen eingehenden Datensatz auf Grund bestimmter Auswahlkriterien verwerfen oder auch passieren lassen.

Eine weitere wichtige Aufgabe des Filters ist es, Datensätze, die von einem (beliebigen) GIDS-Agenten in die Datenbank geschrieben wurden, auszufiltern. Eine lokale GIDS-Datenbank wird zum einen mit Informationen der bei einem Ressourcenabieter lokal installierten Agenten gefüllt, zum anderen werden Daten vom GIDS-Bus durch den GIDS-

Agenten in die Datenbank geschrieben. Werden nun *alle* neu in die GIDS-Datenbank eingefügten Datensätze wieder veröffentlicht, so wird es zwangsläufig zu Duplikaten in den Datenbeständen und somit zu Endlosschleifen der Nachrichten kommen, wodurch in kürzester Zeit eine Überlastsituation (Netzkapazitäten, Speicherkapazität der Datenbanken, Informationsflut für analysierenden IDS-Instanzen etc.) im GIDS zustande kommen würde.

Aggregator/Verdichter. Diejenigen Datensätze, die nicht zuvor durch den Filter aus dem Informationsstrom entfernt worden sind, können in dieser Komponente aggregiert oder verdichtet werden. Eine Aggregation (auch Konsolidierung oder Verdichtung) bezeichnet das Zusammenfassen vieler Daten mit wenig Informationen zu wenigen Daten mit entsprechend hohem Informationsgehalt. Für eine Aggregation wird eine Aggregationsfunktionen benötigt, die zum Beispiel im Falle einer Menge von Zahlen der Mittelwert, das Minimum, das Maximum oder die Summe sein können. Im Fall eines Grid-basierten IDS bietet es sich an, einzelne, fehlgeschlagene Login-Versuche zu einem Angriff zu aggregieren. Neben der Datenkompression wird der eigentliche brute-Force Angriff hervorgehoben, ohne die wesentlichen Informationen zu beeinträchtigen. Durch den Schritt der Aggregation kann also eine Datenverdichtung und Fokussierung erfolgen und somit das Aufkommen und Ungenauigkeiten von Informationen nochmals deutlich gesenkt werden.

Anonymisierer/Pseudonymisierer. Bevor die (aggregierten) Datensätze die administrativen Grenzen eines GIDS-Teilnehmers verlassen, müssen neben den Informationsverbreitungsrichtlinien, die durch die Filterkomponente gewahrt worden sind, auch Datenschutzbestimmungen eingehalten werden. Insbesondere rechtliche Randbedingungen zwingen einen Ressourcenanbieter unter anderem dazu, keine personenbezogenen Daten nach außen zu tragen, was durch den Vorgang der Anonymisierung oder einer Pseudonymisierung gewährleistet wird.

Die Reihenfolge, in der alle Informationen aus der Datenbank die drei zuvorstehenden Komponenten durchlaufen, ist prinzipiell für die Funktionalität nicht entscheidend. Bei der Anordnung dieser Komponenten wird aus Effizienzgründen an erster Stelle eine Filterung (also Löschung „unerwünschter“ Datensätze), dann eine Informationsverdichtung (also eine nochmalige Datenreduktion) und erst abschließend eine Anonymisierung bzw. Pseudonymisierung vorgenommen.

Alle notwendigen Komponenten zum Aufbau des Grid-basierten Intrusion Detection Systems werden in den nachfolgenden Unterabschnitten genauer beleuchtet sowie mögliche Techniken zu deren Umsetzung kurz angesprochen.

3.1.1 Agenten und zentraler Datenspeicher

Agenten dienen in erster Linie dazu Informationen, die für ein GIDS von Interesse sind, semantisch und syntaktisch anzupassen und in einem zentralen Datenspeicher (zum Beispiel einer Datenbank) abzuspeichern. Diese Aufgabe wird typischer Weise unter Nutzung des *Adapter Pattern* (auch bekannt als *Wrapper Pattern*) [4] realisiert.

Bereits an dieser Stelle wird klar, dass zwingend ein einheitliches Daten- und Informationsmodell für das gesamte GIDS notwendig ist. Da prinzipiell Agenten für eine beliebige Datenquelle implementiert werden können, muss insbesondere eine Möglichkeit der Erweiterbarkeit des Modells gegeben sein. Ein mögliches Datenmodell wird im RFC 4765 mit dem *Intrusion Detection Message Exchange Format (IDMEF)* [10] spezifiziert. IDMEF hat sich bereits während einer ungewöhnlich langen Phase des Daseins als Internet Draft recht weit verbreitet. Es handelt sich bei diesem Nachrichtenformat um ein XML-Schema, das alle zuvor erwähnten Eigenschaften realisiert.

Für den Einsatz von IDMEF innerhalb von GIDS müssen am Format aber einige Anpassungen durchgeführt werden. Diese sind notwendig, da IDMEF unter anderem keine Möglichkeit vorsieht, kenntlich zu machen, ob eine IDMEF-Nachricht aus datenschutzrechtlichen Gründen pseudonymisiert wurde. Diese und andere Anpassungen wurden in [9] durchgeführt und können dort nachgeschlagen werden. Bei allen Anpassungen musste jedoch darauf geachtet werden,

dass die Änderungen die Interoperabilität zu anderen Produkten, die das IDMEF-Format unterstützen, nicht gefährden. Aus diesem Grund wurden im wesentlichen Wertebereiche von Freitextfeldern auf bestimmte, im GIDS-Kontext verwendbare Werte eingeschränkt oder es wurden im IDMEF-Standard vorgesehene Methoden zur Erweiterung und Ergänzung des Standards verwendet.

Für einen Domänen-zentralen Datenspeicher bietet sich ein Datenbanksystem an. Dieses arbeitet hoch optimiert und effizient und ist dadurch in der Lage das vergleichsweise hohe Datenvolumen, das an einer Domäne zu erwarten ist, zu verarbeiten. Ein weiterer Vorteil einer Datenbank ist, dass alle angefallenen Daten hierin sinnvoll archiviert werden können und zum Zwecke des Reporting und der Forensik auch zu einem späteren Zeitpunkt noch Abfragen getätigt werden können. Hierzu ist natürlich insbesondere ein Zeitstempel innerhalb eines jeden Datensatzes notwendig, um Abfragen sinnvoll gestalten zu können. Auch für eine Archivierung der Daten ist eine Zeitangabe notwendig.

Es bleibt für jede Site spezifisch einen Prozess zu spezifizieren, der entscheidet, welche Datensätze über welchen Zeitraum aufbewahrt werden dürfen und sollen. Außerdem ist es denkbar, ältere Daten unter Berücksichtigung des geltenden Rechtes in einer höher aggregierten, durchaus mit Informationsverlust behafteten, Version zu speichern, bevor es zur endgültigen Löschung kommt.

3.1.2 Filter

Jeder neu in die Datenbank eingefügte Datensatz muss zeitnah im GIDS veröffentlicht werden. Dazu müssen alle neu in die Datenbank eingefügten Datensätze an den Filter weitergegeben werden, der die erste Komponente in den Verarbeitungsschritten bis hin zur endgültigen Veröffentlichung der Information ist. Optimal ist hierzu ein Push-Verfahren zur Kommunikation, bei dem die Datenbank nach erfolgreichem Einfügen eines Datensatzes diesen direkt an den Filter weitergibt. Sollte dies im Zuge der Implementierung des Systems nicht möglich sein, so kann alternativ ein Pull-Verfahren seitens des Filters realisiert werden. In diesem Fall muss der Filter in regelmäßigen Abständen eine Anfrage an die Datenbank nach neu eingetroffenen Datensätzen richten. Die Intervalle, in denen eine erneute Abfrage stattfindet, müssen individuell an das Szenario angepasst werden, sollten aber in jedem Fall im Bereich von Sekunden liegen. Für eine solche Pull-Kommunikation ist natürlich ein Zeitstempel der Einfügung eines neuen Datensatzes in die Datenbank zwingend notwendig, um eine intervallbasierte Abfrage realisieren zu können.

Wie bereits in [8] erwähnt, ist es eine der Hauptaufgaben des Filters keinen Datensatz, der von einem GIDS-Agenten in die Datenbank geschrieben wurde, wieder zur Neuveröffentlichung freizugeben. Dies ist notwendig, um Endlosschleifen beim Schreiben von Datensätzen im GIDS zu unterbinden. Egal ob eine Push- oder Pull-Kommunikation zwischen der Datenbank und dem Filter etabliert wird, um Datensätze, deren Ursprung ein GIDS-Agent ist, filtern zu können, bedarf es eines entsprechenden Datenfeldes. Eine nicht notwendige, aber dennoch sinnvolle Einrichtung ist hierzu ein Datenfeld, das die Herkunft einer Nachricht vermerkt. Die Herkunftsangabe kann natürlich in einer anonymisierten oder pseudonymisierten Form vorliegen, es reicht prinzipiell jedoch eine binäre Markierung, ob ein Datensatz von einem GIDS-Agenten geschrieben wurde oder nicht.

Eine weitere Aufgabe des Filters ist die (technische) Durchsetzung der Informationsverbreitungsrichtlinien, die spezifisch je Teilnehmer geregelt sind. Im Allgemeinen bietet sich hier die Nutzung regulärer Ausdrücke an. Im Falle der Nutzung eines XML-basierten Datenaustauschformates kann alternativ auch die *Extensible Stylesheet Language* (XSL) [25] und eine *XSL Transformation* (XSLT) zum Einsatz kommen. Hierfür stehen eine Reihe an Implementierungen bereits zur Verfügung.

Durch die Festlegung auf eine Variation von IDMEF als Datenaustauschformat für GIDS und somit auf ein XML-basiertes Datenaustauschformat wird für den Filter voraussichtlich XSL in Verbindung mit XSLT vorbehaltlich einer Durchsatz- und Performanzmessung zum Einsatz kommen. In jedem Fall wird dadurch den Ressourcenanbietern eine sehr restriktive Möglichkeit in die Hand gegeben, ihre „Information Sharing Policies“ durchzusetzen.

3.1.3 Aggregator/Verdichter

Nachdem eine Vorauswahl potentiell im GIDS zu veröffentlichender Informationen durch die vorgeschaltete Komponente des Filters getroffen worden ist, übernimmt der Aggregator oder auch Verdichter eine weitere Datenreduktion. Eine solche Reduktion ist vom Konzept her zwischen einer verlustfreien und einer verlustbehaftete Datenverdichtung zu unterscheiden. Ein Beispiel für eine verlustfreie Verdichtung ist, wenn es das Ziel ist über ein zuvor bekanntes Zeitintervall die Anzahl an übertragenen Paketen zu ermitteln. In diesem Fall könnte der Aggregator als Aggregationsfunktion eine Summierung aller übertragenen Pakete über eben dieses bekannte Zeitintervall vornehmen, was keinen Informationsverlust für dieses spezielle Auswertungsverfahren bedeuten würde. Einen Informationsverlust im Kontext dieses Beispiels müsste man hingegen bei der Aggregatsbildung über z.B. das doppelte Zeitintervall hinnehmen.

Bereits dieses triviale Beispiel verdeutlicht, dass der Verdichter in jedem Fall einen Puffermechanismus für eintreffende Daten haben muss. Er erhält seine Eingabedaten asynchron per Push-Kommunikation von der Komponente des Filters, aggregiert die eingehenden Daten entsprechend seiner Spezifikationen und leitet die verdichteten Datensätze per Push-Kommunikation weiter an den Anonymisierer und Pseudonymisierer. Zweitere Kommunikationsbeziehung kann sowohl zeitgesteuert synchron als auch asynchron, durch bestimmte Ereignisse gesteuert, erfolgen. Z.B. ist es denkbar, dass alle x Minuten in jedem Fall ein Informationsaggregat ausgegeben wird, im Falle einer eintreffenden Alarmmeldung der lokalen (G)IDS-Instanz diese jedoch sofort weitergeleitet wird.

3.1.4 Anonymisierer und Pseudonymisierer

Insbesondere zur Wahrung von Datenschutzrichtlinien ist eine Anonymisierung und/oder Pseudonymisierung von Informationen vor ihrer Veröffentlichung in einem Grid-weit föderierten System zwingend notwendig. Sowohl die Anonymisierung als auch die Pseudonymisierung sind Maßnahmen zur Wahrung des Datenschutzes. Mit der *Anonymisierung* bezeichnet man das Verändern personenbezogener Daten, so dass diese Daten im Nachhinein nicht mehr eindeutig einer Person zugeordnet werden können. In Abgrenzung zur Anonymisierung bezeichnet man mit dem Vorgang der *Pseudonymisierung* eines Datensatzes das Ersetzen der Identifikationsmerkmale, die eine Zuordnung der Daten zu einer Person ermöglichen, durch ein Pseudonym. Die Bildung der Pseudonyme geschieht hierbei durch eine Abbildung (in der Regel bijektiv), so dass unter Kenntnis der Abbildungsfunktion, die auch als *Code* bezeichnet wird, eine Zuordnung eines Datensatzes zu einer Person eindeutig möglich ist. Im Gegensatz zur Anonymisierung bleiben bei der Pseudonymisierung Zusammenhänge unterhalb der pseudonymisierten Datensätze erhalten, unter der Voraussetzung, dass alle Datensätze durch dieselbe Abbildung pseudonymisiert worden sind.

Die Komponente des Anonymisierers und Pseudonymisierers bezieht als Eingabe seine Informationen asynchron, per Push-Kommunikation vom Aggregator. Als Ausgabe liefert er anonymisierte und/oder pseudonymisierte Datensätze im selben Datenformat wie die zuvor erhaltenen Eingabedaten. In diesen bereinigten Datensätzen befinden sich entsprechend nur noch mit den Datenschutzrichtlinien konforme Informationen. Es ist zweckmäßig sich Gridweit auf eine einheitliche Anonymisierungs- oder Pseudonymisierungsfunktion festzulegen, um eine höhere Qualität der Angriffserkennung gewährleisten zu können, insbesondere im Falle der Pseudonymisierung (s.o.).

Für eine Implementierung kann eventuell eine Anlehnung an das *Secure Audit Logging for Linux (SAL)* Projekt [22], *SNARE (System iNtrusion Analysis & Reporting Environment)* [23] oder auch *The Linux Basic Security Module Project (Linux BSM)* [1] in Betracht gezogen werden. Alle diese Projekte behaupten von sich konform zu den *Common Criteria for Information Technology Security Evaluation* (auch abgekürzt als *Common Criteria* oder *CC*) zu sein, die in dem internationalen Standard ISO/IEC 15408 [11, 12, 13] niedergeschrieben sind und ein Äquivalent zu dem *U.S. Government's C2 standards for security* [3] (auch bekannt als „*The Orange Book*“) bilden. Im Falle der Nutzung eines XML-basierten Datenaustauschformates kann auch hier wie bereits beim Filter erwähnt die *Extensible Stylesheet Language* (XSL) [25] und eine *XSL Transformation* (XSLT) zum Einsatz kommen.

Die Komponente *Anonymisierer und Pseudonymisierer* dient in erster Instanz dazu, dass die juristischen und vor allem datenschutzrechtlichen Randbedingungen, denen GIDS in Deutschland unterliegt, auch technisch durchgesetzt werden können. Für eine detaillierte Betrachtung des Datenschutzes und des verfassten Datenschutzkonzeptes sei an dieser Stelle auf Meilenstein 13 [7] verwiesen.

3.1.5 GIDS-Agent

Die GIDS-Agenten sind die Komponenten, die die Kommunikation, also den Informationsaustausch, unterhalb aller teilnehmenden Sites realisieren. Zu den Hauptaufgaben eines GIDS-Agenten gehört unter anderem die Authentifizierung und Autorisierung der anderen GIDS-Agenten sowie die Gewährleistung der Nachrichtenintegrität, Nachrichtenvertraulichkeit und Authentizität.

Für die Umsetzung der GIDS-Agenten bietet sich die Verwendung des *Observer Pattern* (auch bekannt als *Publish-Subscribe Pattern* oder *Dependents Pattern*) [4] an. Das Observer Pattern definiert dabei eine $1 : n$-Beziehung zwischen den GIDS-Agenten. Es ist dazu intendiert, dass, wenn sich der Zustand eines der Objekte ändert, alle anderen Objekte automatisch über diese Änderung informiert werden. Im Fall der GIDS-Agenten bedeutet dies, dass, wenn einer der GIDS-Agenten einen Datensatz zur Informationsverbreitung erhält (dies entspricht also der Zustandsänderung), so informiert er alle weiteren GIDS-Agenten über den Eingang dieses Datensatzes. Einen möglichen Ansatz zur Einführung des Publisher-Subscriber Design Pattern in Infrastrukturen verteilter IDS beschreiben Basicevic et. al. in [2]. Bei der Implementierung der GIDS-Agenten kann weiter das bereits existierende und wohlbekannte Projekt *Prelude* [15] als Basis dienlich sein, welches sich im weiteren Projektverlauf beweisen wird müssen.

Zur Kommunikation der GIDS-Agenten untereinander bietet sich unter anderem das *The Intrusion Detection Exchange Protocol* (IDXP) [21] an. Unter Nutzung des *Blocks Extensible Exchange Protocol* (BEEP) [17], das auf TCP als unterliegendes Protokoll abgebildet werden kann [18] und durch *Transport Layer Security (TLS)* [20] gesichert wird, kann IDXP die Vertraulichkeit, Integrität und Authentizität von Nachrichten gewährleisten. Weiter entstehen in der Regel keine Komplikationen mit Firewalls durch das Tunneln über TCP bzw. beschränken sich ggf. Änderungen an Firewall-Regeln auf die Freigabe eines TCP-Ports für den GIDS-Agenten. Unter Umständen ist BEEP sogar auch in Kombination mit Network Address Translation (NAT) einsetzbar. Zusätzlich kann BEEP die Duplikation von Nachrichten bei der Kommunikation von GIDS-Agenten untereinander verhindern.

Als Alternative, die im Fall des GIDS-Projekts auch hoch wahrscheinlich implementiert wird, kann ein *Virtuelles Privates Netz* (VPN) zwischen allen GIDS-Agenten etabliert werden, das Multicast-Fähigkeiten besitzt. Hierzu kann ein SSL-basierter Tunnel der ISO/OSI Schicht zwei eingerichtet werden, so das eine IP-basierte Multicat-Fähigkeit direkt gegeben ist. Alle notwendigen Sicherheitsanforderungen an die Kommunikation der GIDS-Agenten untereinander werden dabei durch SSL (*Secure Socket Layer*)bzw. TLS [20] implementiert. Um die Authentifizierung und Autorisierung weiterer GIDS-Agenten durchführen zu können, kann auf eine im Grid bereits bestehende Public Key Infrastruktur zurückgegriffen werden. Diese vorhandene PKI ermöglicht eine einfache Einbindung der GIDS-Agenten in das Gridumfeld. Hierfür werden für die Agenten X.509 Zertifikate [19] zur Verfügung gestellt, wodurch sowohl Authentifizierungs- als auch Autorisierungsfunktionalitäten gegeben sind.

3.1.6 Lokale (G)IDS-Instanz

Die eigentliche Logik des IDS liegt in den angeschlossenen (G)IDS-Instanzen bzw. ihrem Pendant, den globalen GIDS-Instanzen in der Rolle des Betreibers. Diese implementieren eine Analysefunktionalität und sind somit neben der Platzierung der IDS-Sensoren zentraler Bestandteil bei der Erkennung von Angriffen. Kritische Eigenschaften des lokalen (G)IDS-Instanz sind zum einen Performanz (wie viele eingehende Meldungen können pro Zeiteinheit verarbeiten?), zum anderen die Erkennungsleistung (wie hoch ist die Rate der False Positives bzw. False

Negatives?). Diese Eigenschaften werden essentiell sowohl vom Konzept der Analyseeinheit als auch ihrer Implementierung beeinflusst.

Für eine lokale (G)IDS-Instanz kommen prinzipiell alle Arten der Datenanalyse in Betracht: Es können sowohl eine Anomalieerkennung wie auch eine signaturbasierte Angriffserkennung, ein hybrides Verfahren oder sogar mehrere Verfahren parallel zum Einsatz kommen. Die lokalen GIDS-Instanzen diesen vor allem als Ergänzung zu einer globalen Angriffserkennung. Da dem Grid-übergreifenden System stets nur die von den jeweiligen Ressourcenanbietern gefilterten und anonymiserten bzw. pseudonymisierten Informationen zur Auswertung vorliegen, muss die Grid-globale GIDS-Instanz mit einem gegebenen Informationsdefizit auskommen und umgehen können. Lokal installierte GIDS-Instanzen hingegen können zumindest auf die ungefilterten und nicht manipulierten Informationen ihrer eigenen Ressourcen, die in der lokal betriebenen GIDS-Datenbank hinterlegt sind, zurückgreifen und somit in Einzelfällen eine deutlich bessere Erkennungsleistung erzielen. Dieser Umstand ist ebenfalls in mehreren wissenschaftlichen Arbeiten belegt worden [5, 6], dennoch bleibt die Installation und der Betrieb einer lokalen GIDS-Instanz optional und dem jeweiligen Ressourcenanbieter selbstverständlich eigenverantwortlich selber überlassen.

Für eine konkrete Implementierung einer lokalen (G)IDS-Instanz bietet es sich an, auf bereits existierende Mechanismen zurückzugreifen und diese an die Grid-Umgebung anzupassen. So kann zum Beispiel für ein signaturbasiertes Erkennungsverfahren wie *Snort* [24] ein entsprechender Regelsatz neben dem klassischen Regelwerk entworfen und implementiert werden. Für eine Anomalieerkennung gilt es die für das Verfahren notwendigen Parameter an die neue Umgebung anzupassen.

Kapitel 4

Detaillierung der Architektur auf Seiten eines GIDS-Betreibers zur Erbringung eines Grid-weiten IDS-Dienstes

4.1 Architektur auf Seiten des Betreibers des GIDS

Die Architektur auf Seiten des Betreibers des GIDS steht in Anlehnung an den Aufbau des Systems auf Seiten eines Ressourcenanbieters. In Abgrenzung zu einem Ressourcenanbieter stellt der Betreiber des GIDS in der Regel jedoch keine Rohdaten für eine gemeinschaftliche Angriffserkennung zur Verfügung, da er nicht über entsprechende Datenquellen verfügt. Vielmehr stellt er einen Grid-Dienst zur Verfügung, der eine Berichterstattung und Darstellung der im GIDS verfügbaren Informationen und Berichte für die unterschiedlichen Nutzergruppen (Ressourcenanbieter, VOs etc.) anbietet. Entsprechend ist die Architektur auf Seiten des GIDS-Betreibers um die Komponenten zur Datenakquise (die Agenten) und die Komponenten zur datenschutzkonformen Informationsaufarbeitung vermindert. Zur Dienstbereitstellung jedoch werden Anbindungen an Grid-typische Dienste (z.B. VO-Managementsysteme) vorgenommen. Abbildung 4.1 stellt den genauen Aufbau des GIDS auf Seiten seines Betreibers graphisch dar.

GIDS-Agent und Datenspeicher. Sowohl aus Architektur- als auch aus Implementierungssicht ist der GIDS-Agent und der Datenspeicher identisch mit den vergleichbaren Komponenten auf Seiten eines Ressourcenanbieters. Im Falle des Betreibers des GIDS ist es die Hauptaufgabe des GIDS-Agenten den Datenspeicher mit Informationen, die von den anderen Partnern zur Analyse publiziert werden, zu füllen. Zusätzlich versendet er Grid-global erkannte Angriffe, die durch die entsprechende GIDS-Instanz erkannt wurden. In der Regel stellt der GIDS-Agent die einzige Datenquelle für den Betreiber des GIDS dar. Eine mögliche Ausnahme ist in Abschnitt 3.2 in Meilenstein 10 [8] aufgezeigt.

Durch die Gleichheit mit dem GIDS-Agenten und Datenspeicher auf Seiten eines Ressourcenanbieters bedingt wird nachfolgend nicht näher auf diese Komponenten eingegangen. Für eine genauere Beschreibung sei auf den Abschnitt 3 und darin insbesondere das Unterkapitel 3.1.5 verwiesen.

Grid-globale GIDS-Instanz. Der Grid-globalen Instanz zur Auswertung der im Grid verfügbaren Informationen stehen in der Regel die wenigsten Originalinformationen zur Analyse zur Verfügung. Dies liegt daran, dass in der zugehörigen Informationsbasis ausschließlich die im Grid „öffentlich“ verfügbaren Daten, die von den beteiligten GIDS-Agenten publiziert worden sind, verfügbar sind. Diese Informationen sind bereits von

Abbildung 4.1: Architektur auf Seiten des Betreibers des GIDS

den jeweiligen Ressourcenanbietern gefiltert, aggregiert und verdichtet sowie anonymisiert und/oder pseudonymisiert worden (siehe auch Abschnitt 3). Daraus folgt, dass dieser Instanz des IDS nur eine Teilmenge der Informationen vorliegt, die paarweise jedem einzelnen Ressourcenanbieter zur Verfügung steht. Es ist also zu erwarten, dass die Grid-globale GIDS-Instanz in Bezug auf ihre Erkennungsleistung gegenüber den jeweils lokalen (G)IDS-Instanzen schlechter abschneidet. Aus diesem Grund erscheint es sinnvoll, dass auch durch (G)IDS-Instanzen erkannte Angriffe durch die Ressourcenanbieter Grid-global publiziert werden, wie auch konzeptuell in Abschnitt 3 vorgesehen.

Benutzerportal. Auf Basis der in der Datenbank hinterlegten Berichte stellt das Benutzerportal eine mandantenfähige Nutzeroberfläche zur Berichterstattung bereit. Nutzer (Mitglieder einer VO oder auch Ressourcenanbieter) können hier den aktuellen Sicherheitsstatus des Grids unter Nutzung ihrer im Grid gültigen Credentials einsehen sowie historische Berichte anfragen. Es werden verschiedene Sichten auf die Berichte je nach Rolle des Nutzers angeboten.

Da das Benutzerportal auf der einen Seite nur Abhängigkeiten von bestehenden Grid-Diensten und auf der anderen Seite von einer GIDS-Datenbank hat, kann es auch zum Betrieb bei einem beliebigen anderen Teilnehmer des GIDS verwendet werden. Dadurch kann eine redundante Auslegung bzw. eine Wiederverwendung dieser Komponente als Managementoberfläche bei den Ressourcenanbietern ermöglicht werden.

Proaktive Benachrichtigung. Diese Komponente dient dazu die Kunden des GIDS, also sowohl Mitglieder einer VO als auch Ressourcenanbieter, über die aktuelle Sicherheitslage stets proaktiv, d.h. sofort nach erkanntem Angriff, in Kenntnis zu setzen. Dies kann auf verschiedenen Kommunikationswegen wie z.B. E-Mail oder SMS geschehen. Eine Auswahl des Kommunikationskanals kann in Abhängigkeit der Wichtigkeit einer Nachricht erfolgen.

Genau wie für das Nutzerportal auch, bestehen für die Komponente der proaktiven Benachrichtigung nur Abhängigkeiten zu bestehenden Grid-Diensten und einer GIDS-

Datenbank. In hiesigem Fall ist die Anbindung an eine Datenbank zur Abbildung von Ressourcen auf die sie nutzenden VOs sowie an das VO-Managementsystem zum Bezug der Kontaktdaten der jeweils verantwortlichen Personen notwendig. Somit kann auch diese Komponente bei einem beliebigen anderen Teilnehmer des GIDS betrieben werden.

4.1.1 Grid-globale IDS-Instanz

Die beim Betreiber des GIDS zum Einsatz kommende Grid-globale IDS-Instanz ist in erster Linie identisch mit einer jeden lokalen (G)IDS-Instanz (vgl. Abbildung 3.1 und Abschnitt 3.1.6). Ein entscheidender Unterschied zwischen den beiden Instanziierungen dieser Komponente ist die Datenbasis, auf der sie arbeiten. Während einer lokalen (G)IDS-Instanz alle im Grid anonymisierten und/oder pseudonymisierten Daten zusätzlich zu den unveränderten Rohdaten der eigenen administrativen Domäne zur Auswertung zur Verfügung stehen, kann die Grid-globale IDS-Instanz ausschließlich auf den anonymisierten und/oder pseudonymisierten Datenbestand der an GIDS beteiligten Partner zurückgreifen. Diese Tatsache lässt eine geringere Erkennungsleistung auf Grund von Informationseinbußen an dieser Stelle erwarten. Zudem kann eine individuelle Anpassung der zum Betrieb der Grid-globalen IDS-Instanz notwendigen Regelwerke bzw. Parameter in Abhängigkeit von der Qualität des Informationsbestandes erforderlich sein.

Die Hauptmotivation für den Betrieb einer Grid-globalen IDS-Instanz ist zweischichtig. Zum einen steht hierbei der Dienstgedanke im Vordergrund. Der Betrieb eines IDS-Dienstes für das D-Grid ermöglicht es Kunden, die keine Ressourcenanbieter sein müssen, einen Überblick über die aktuelle Sicherheitslage im und des D-Grid zu vermitteln. Zum anderen stehen natürlich die strategischen Vorteile einer Grid-globalen Ereigniskorrelation und somit einer in einigen Fällen schnelleren oder auch besseren Erkennung verteilt angelegter Angriffe im Vordergrund. Diese Eigenschaft verteilt eingerichteter IDS-Instanzen ist wissenschaftlich zum Beispiel in [5, 6] belegt und untermauert.

4.1.2 Benutzerportal

Das Benutzerportal stellt eine zentrale Anlaufstelle für die Kunden des GIDS bereit, in dem kundenspezifische Sichten auf die verfügbaren Reporte realisiert werden. Zur Nutzerauthentifizierung ist eine Anbindung an bestehende AA-Infrastrukturen bzw. VO-Managementsysteme notwendig, die in Abbildung 4.1 grau als bereits existierende Systeme im Grid dargestellt sind. Eine Einschränkung des Nutzerkreises auf Grid-Nutzer ist insofern sinnvoll, als dass einem externen Angreifer kein Vorteil durch die Einsicht der verfügbaren GIDS-Berichte entstehen soll und er so evtl. Rückschlüsse auf bestehende Sicherheitslücken, Schwachstellen o.ä. schließen könnte.

Alle in diesem Portal verfügbaren Informationen sind durch ihre zuvor datenschutzkonforme Aufarbeitung prinzipiell für alle Anwender im Grid einsehbar. Es wird jedoch zusätzlich eine Sicht auf die Berichte angeboten, die nur die eigenen Ressourcen bzw. die von einer VO verwendeten Ressourcen umfasst. Dazu ist eine Abbildung von Ressourcen zu VOs notwendig. Eine solche Datenbank kann z.B. Bestandteil eines Grid-Monitoring-Systems sein wie u.a. im Falle des D-Grid. In jedem Fall kann hier wie bei der Nutzerauthentifikation auf einen bestehenden Dienst im Grid zurückgegriffen werden, weswegen die entsprechende Datenbank in Abbildung 4.1 ebenfalls grau schattiert dargestellt ist.

Zur Implementierung des Benutzerportals eignet sich beispielsweise eine Orientierung an dem sogenannten *Bridge Pattern* nach [4]. Hierdurch kann erreicht werden, dass die Abstraktion des Benutzerportals von seiner Implementierung vollkommen losgelöst ist. Im Gegensatz zu regulären Vererbungen wird durch den Einsatz des Bridge Pattern eine harte Bindung von Abstraktion und Implementierung vermieden, so dass sowohl die Abstraktion, als auch die Implementierung durch Unterklassen erweiterbar bleiben, während dies keinen Einfluss auf nachgelagerte Anwendungen hat.

4.1.3 Proaktive Benachrichtigung

Für eine proaktive Benachrichtigung der Kunden des Grid-basierten IDS ist eine mandantenfähige Komponente notwendig, mit deren Hilfe Benachrichtigungen über einen erkannten Sicherheitsvorfall abhängig von seinem Schweregrad (engl. *severity*) über verschiedene Kommunikationswege, wie beispielsweise E-Mail oder SMS, versendet werden. Ein jeder Nutzer hat die Möglichkeit sich die Art der Benachrichtigung an seine eigenen Bedürfnisse anzupassen, wobei ein Ausbleiben einer Benachrichtigung auch eine zulässige Aktion ist. Zur Nutzerauthentifizierung und -autorisierung ist hierzu eine Anbindung an die AA-Infrastruktur als Teil des VO-Management (siehe grau schattierte Komponente in Abbildung 4.1) notwendig. Eine weitere Anbindung an eine Datenbank, die die im Grid verfügbaren Ressourcen auf die sie nutzenden VOs und umgekehrt abbildet, ist notwendig, um Benachrichtigungen selektiv an die sie jeweils betreffenden VO-Verantwortlichen zu übermitteln.

Für eine Implementierung dieser Komponente bietet sich das *Observer Pattern* (auch als *Dependents* oder *Publish-Subscribe Pattern* bekannt) nach [4] an. Dieses definiert eine „1-zu-viele“ (engl. *one-to-many*) Beziehung zwischen Objekten. Im Falle einer Statusänderung eines Objektes werden die anderen am System beteiligten Objekte über eben diese Änderung in Kenntnis gesetzt. Dieses Prinzip lässt sich somit einsetzen um die tatsächlich benachrichtigenden Teilkomponenten (z.B. ein E-Mail- oder SMS-Benachrichtigungsmodul) über neu eingehende Sicherheitsvorfälle in Kenntnis zu setzen.

Um eine proaktive Benachrichtigungskomponente zu realisieren, bietet sich unter anderem der Einsatz bzw. die Modifikation bereits existierender Monitoring-Programme an. Es ist zum Beispiel denkbar, das Überwachungswerkzeug *Nagios* [14] für solche Zwecke zu modifizieren. Nagios arbeitet mit Plug-Ins, die bei dem Eintritt eines Ereignisses (z.B. die Überschreitung eines Schwellwertes) eine Meldung generieren. Der Konfiguration von Nagios entsprechend wird auf Grund dieses Ereignisses ggf. eine Benachrichtigung eines zuvor konfigurierten Nutzerkreises über einen zuvor ebenfalls festgelegten Verbreitungsweg vorgenommen. Zusätzlich bietet Nagios eine Benutzeroberfläche, in der der aktuelle Status der überwachten Ressourcen grafisch dargestellt wird. Natürlich sind für solche Zwecke auch andere Monitoring-Lösungen wie beispielsweise die quelloffenen Produkte/Projekte *Cacti* (`http://www.cacti.net/`) oder auch *Zabbix* (`http://www.zabbix.com/`) prinzipiell denkbar. Bei kommerziellen Produkten wie z.B. *IBM Tivoli* oder *HP OpenView* bleibt die Einsatzmöglichkeit auf Grund ihrer nur bedingt anpassbaren Funktionsweise zu prüfen.

Kapitel 5

Zusammenfassung

Ziel dieses Dokuments ist die Spezifizierung der Architektur, die dem GIDS zu Grunde liegt. Grundlage für die Erstellung des Architekturentwurfs bilden die Anforderungen, die in Meilenstein 6 (*MS6* [16]) erhoben wurden. Dieses Dokument verfeinert entsprechend die bereits in Meilenstein 10 (*MS10* [8]) abgeleitete Grobarchitektur. Als Konsequenz des föderativen Charakters des GIDS-Konzepts und den darin vertretenen Rollen ist eine technische sowie administrative Unabhängigkeit dieser einzelnen Teilnehmer des GIDS zu gewährleisten.

Die grundlegende Idee ist die Trennung in einen jeweils site-lokalen Bereich, in dem die Partner organisatorisch und administrativ unabhängig agieren könnnen. Dieser beinhaltet die technischen Sensoren bzw. Agenten zur Erhebung von IDS-relevanten Rohdaten und der Erkennung von Angriffen, auf die nur der entsprechende GIDS-Partner direkten Zugriff hat. Eine detaillierte Beschreibung der Architektur auf Seiten eines Ressourcenanbieters ist in Kapitel 3.1 zu finden.

Weiterhin ist ein globaler GIDS-Dienst vorgesehen, der den Export der Daten und eine globale Analyseeinheit und Reporting-Komponente beinhaltet. Konzeptionell sind die beiden Bereiche gleich aufgebaut und durch den GIDS-Bus miteinander verbunden, jedoch sind die Daten im globalen Bereich für jeden Partner zugreifbar. Zudem bietet ein GIDS-Dienstanbieter allen D-Grid-Teilnehmern die Möglichkeit, sich einen Überblick über die aktuelle Sicherheitslage im D-Grid zu verschaffen. Ein genauer Überblick über die Architektur auf Seiten des Betreibers des GIDS ist in Kapitel 4.1 zu finden.

Abbildungsverzeichnis

Literaturverzeichnis

[1] Jeremy Banford. The linux basic security module project (linux bsm), November 2000.

[2] Ilija Basicevic, Miroslav Popovic, and Vladimir Kovacevic. Use of publisher-subscriber design pattern in infrastructure of distributed ids systems. In *ICNS '07: Proceedings of the Third International Conference on Networking and Services*, Washington, DC, USA, 2007. IEEE Computer Society.

[3] Department of Defense Trusted Computer System Evaluation Criteria, DoD 5200.28-STD, December 1985.

[4] Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series, November 1994.

[5] N. gentschen Felde. Einsatz der graphbasierten Meldungsstrukturanalyse in domänenübergreifenden Meta-IDS. In *Lecture Notes in Informatics — Informatik 2005, Informatik LIVE!*, number P–68 in Band 2, pages 653–657, Bonn, Germany, September 2005. Gesellschaft für Informatik.

[6] N. gentschen Felde, M. Jahnke, P. Martini, and J. Tölle. Impact of Sanitized Message Flows in a Cooperative Intrusion Warning System. In *Proceedings of the 25th Military Communications Conference (MILCOM 2006)*, volume 2006, pages 1–7, Washington, DC, USA, October 2006.

[7] Wolfgang Hommel, Nils gentschen Felde, Felix von Eye, Jan Kohlrausch, and Christian Szongott. Datenschutzmodell für ein Grid-basiertes IDS. Meilensteinbericht, D-Grid, July 2010.

[8] Wolfgang Hommel, Nils gentschen Felde, Felix von Eye, Jan Kohlrausch, and Christian Szongott. Grobskizze einer Architektur. Meilensteinbericht, D-Grid, April 2010.

[9] Wolfgang Hommel, Nils gentschen Felde, Felix von Eye, Jan Kohlrausch, and Christian Szongott. Informationsmodell inklusive Datenaustauschformat. Meilensteinbericht, D-Grid, June 2010.

[10] IETF. The intrusion detection message exchange format (idmef). *http://tools.ietf.org/html/rfc4765*, 2007.

[11] Evaluation criteria for IT security – Part 1: Introduction and general model, October 2005.

[12] Evaluation criteria for IT security – Part 2: Security functional requirements, October 2005.

[13] Evaluation criteria for IT security – Part 3: Security assurance requirements, October 2005.

[14] Nagios – Network Monitoring.

[15] Prelude technologies. https://dev.prelude-technologies.com/.

[16] Helmut Reiser, Nils gentschen Felde, Felix von Eye, Jan Kohlrausch, and Christian Szongott. Anforderungs- und Kritierienkatalog (MS 6). Meilensteinbericht, D-Grid, January 2010.

[17] The blocks extensible exchange protocol core, March 2001.

[18] Mapping the beep core onto tcp, March 2001.

[19] Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile, April 2002.

[20] The transport layer security (tls) protocol, April 2006.

[21] The intrusion detection exchange protocol (idxp), March 2007.

[22] Secure audit logging for linux (sal) – software design document, February 2003.

[23] Snare (system intrusion analysis & reporting environment).

[24] Snort – The Open Source Network Intrusion Detection System.

[25] W3C - World Wide Web Consortium. *Extensible Stylesheet Language (XSL)*, December 2006. `http://www.w3.org/TR/xsl/`.