Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Ausarbeitung AW1

André Harms

Simulation von Cyberangriffen

Fakultät Technik und Informatik
Studiendepartment Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Inhaltsverzeichnis

1 Einführung

In den letzten Jahren, in denen Informationssysteme in nahezu allen Lebensbereichen eingesetzt werden, sind neben Cyberkriminalität auch ernst zu nehmende Cyber-Attacken Realität geworden (John J. Kelly, 2008). So werden kritische Infrastrukturen von Ländern oder wichtige wirtschaftliche Einrichtungen mittlerweile Opfer von zielgerichteten Angriffen. Die wohl bekanntesten Beispiele für ein solches Vorgehen sind ein politisch motivierter Angriff auf Estlands digitale Infrastruktur im Jahre 2007 (Arquilla, 2011) (John J. Kelly, 2008) und der Einsatz von Stuxnet, der das iranische Atomprogramm zum Angriffsziel hatte (Symantec Corporation, 2011).

1.1 Motivation

Dass eine Attacke durch mehrere Angriffsvektoren realisiert werden kann und dass dies durchaus sinnvoll ist, hat vor allem Stuxnet bewiesen. Es nutzt mehrere Wege zu seiner unbemerkten Verbreitung und bedient sich dabei mehrerer Sicherheitslücken und Softwarefehler, um das eigentliche Ziel zu erreichen (Symantec Corporation, 2011). Generell ist die Verbreitung von Schadsoftware und das Angreifen von IT-Systemen ein facettenreicher Prozess. Für das Opfer kommt dabei der Angriff meist unvorhergesehen und ist im Vorwege und auch im Nachhinein nicht oder nur schwer nachzuvollziehen.
Möchte man ein Sicherheitskonzept oder den aktuellen Sicherheitsstand eines Systems testen, kann dies zum Beispiel mittels Penetration- und Vulnerability-Tests geschehen. Diese kommen echten - aber abgesprochenen - Angriffen gleich. Somit besteht auch die Gefahr, ein System unbeabsichtigt bei so einem Test zu beschädigen und einen Ausfall hervorzurufen (Messner, 2011). Diese Tatsache birgt vor allem beim Testen von kritischer Infrastruktur - aufgrund ihrer Bedeutung - Gefahren. Zudem lassen sich mit Penetrationtests nicht unbedingt alle Teile des zu testenden Systems untersuchen. Daher bieten sich Simulationen von Angriffen an, um Erkenntnisse für präventive Maßnahmen zu erlangen.

2 Überblick

2.1 Angriffstypen

Angriffe auf IT-Systeme können auf verschiedenen Ebenen stattfinden. Bei einer genauen Betrachtung lassen sich drei Ebenen klassifizieren:

Internet/Netzwerk: Der Angriff zielt auf das Netzwerk und seine Infrastruktur ab. Dazu gehören generelles Eindringen in Netzwerke (z.B. WLAN) oder das Abfangen und Verändern von Nachrichten (Protokollangriffe). Auch das Provozieren von Ausfällen ist möglich. Beliebte Angriffstechniken sind zum Beispiel DDoS Attacken, die einen Dienst oder Server überlasten und ihn so sabotieren.

Anwendungsebene: Hier werden Sicherheitslücken, Architektur- und Programmierfehler in der verwendeten Software ausgenutzt, um einen erfolgreichen Angriff durchzuführen. Dabei können alle Softwarekomponenten des Systems - wie Betriebssystem, Dienste, Anwendungssoftware - betroffen sein. Zu möglichen Angriffsvektoren gehören hier zum Beispiel Puffer- und Integerüberläufe, Formatstringangriffe und Ausnutzen von Logikfehlern (Erickson, 2009).

Anwenderebene: Sicherheitsverstöße auf Anwenderebene geschehen absichtlich oder unbewusst. Vor allem unbewusste Aktionen stellen ein hohes Sicherheitsrisiko dar. Sie werden von Unwissenheit, Unachtsamkeit und Leichtgläubigkeit des Anwenders begünstigt (Kevin D. Mitnick, 2002). Auch bestimmtes Benutzerverhalten trägt zur Verteilung von Schadsoftware bei und ermöglicht erst einige Angriffsvektoren (z.B. Phishing). Die Sorge um nachlässige Mitarbeiter ist in Unternehmen zwar zurückgegangen, dennoch wird der Mensch als eine der größten Schwachstellen der IT-Sicherheit empfunden, weswegen immer mehr Unternehmen in Sensibilisierungs-Maßnahmen investieren (Capgemini, 2009).

Für viele Angriffsszenarien werden Angriffsmöglichkeiten aus den genannten Teilbereichen kombiniert. So werden zum Beispiel DDoS Attacken häufig erst durch Verwendung eines Bot-

netzes[1] möglich, das wiederum durch das Ausnutzen von Angriffen auf die Anwender- und Anwendungsebene etabliert werden kann (John J. Kelly, 2008).
Den Zusammenhang zwischen den einzelnen Ebenen zeigt Abbildung 2.1. Hier ist zu erkennen, welche Ebenen sich gegenseitig bei Angriffen beeinflussen können und welche Ebene einer anderen gegenüber exponiert ist.

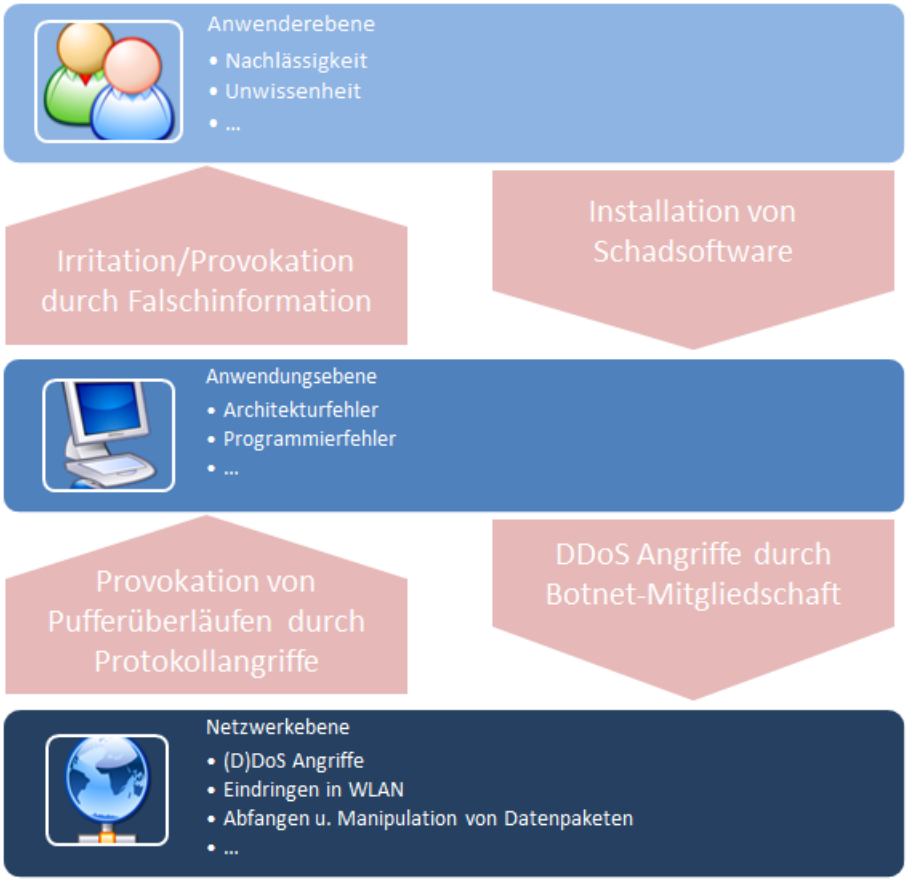

Abbildung 2.1: Potentielle Angriffsebenen mit Schwachstellen und exemplarischen Möglichkeiten zur gegenseitigen Beeinflussung

Um das Zusammenspiel der Ebenen zu verdeutlichen, wird an dieser Stelle das schon erwähnte DDoS-Beispiel konkretisiert:
Ein Angreifer möchte einen Dienst stören. Da der Dienst keine bekannten Fehler oder Sicherheitslücken aufweist, soll der Dienst gestört werden, indem er durch massive Anfragen überfordert wird. Hierzu benötigt der Angreifer mehrere Systeme, die koordiniert eine ausreichende Menge

[1] Zusammenschluss mehrerer gekaperter Systeme

an Anfragen stellen können. Um dies zu erreichen, bietet sich das Etablieren eines Botnetzes an, was vorraussetzt, dass Programmcode des Angreifers auf den unfreiwilligen Teilnehmern des Botnetzes ausgeführt werden kann. Eine Möglichkeit dies zu realisieren, ist das Ausnutzen von Pufferüberläufen aus der Ferne über das Netzwerk (Erickson, 2009). Sollte das Zielsystem keinen für Pufferüberläufe anfälligen über das Netzwerk erreichbaren Dienst ausführen, muss der Angreifer einen anderen Weg finden, seinen Code auf dem System zur Ausführung zu bekommen. Eine Möglichkeit dafür ist das Versenden von E-Mails mit entsprechendem Anhang (ausführbare Datei, manipuliertes PDF-Dokument, ...). Ist so eine Mail entsprechend gut verfasst und motiviert den Benutzer zum Aufrufen des schadhaften Anhangs, besteht die Chance, dass der Empfänger unbedacht mit dem Anhang umgeht und diesen öffnet. Wird nun der Programmcode des Angreifers auf dem Opfersystem ausgeführt, kann es zum Mitglied des Botnetzes gemacht werden. Abschließen kann das System für den ursprünglich geplanten Angriff auf den zu störenden Dienst missbraucht werden.
Aus diesem einfach beschriebenen Szenario wird klar, dass ein Angreifer nicht immer unbedingt den direkten Weg bei einer Attacke beschreiten kann. Es kann unter Umständen nötig sein, alle vorgestellten Ebenen zu verwenden/zu kompromittieren, um einen geplanten Angriff erfolgreich durchführen zu können.

2.2 Simulationen

Zur Zeit existieren mehrere Simulationsansätze, die eine Einschätzung der Ausbreitungsgeschwindigkeit von Schadsoftware oder aber auch die Beobachtung von Angriffen zum Entwickeln von Gegenmaßnahmen ermöglicht. Eine Auswahl grundsätzlicher Vorgehensweisen wird folgend aufgeführt.

2.2.1 Mathematische Modelle

Mittels mathematischer Modelle, die durch Differentialgleichungen beschrieben werden, ist es möglich, die Ausbreitungsgeschwindigkeit von Schadsoftware vorherzusagen. Unter anderem wird dabei auf Modelle der medizinischen Epidemiologie zurückgegriffen (Lora Billings, 2002). Sie arbeiten mit statistischen Werten zu Kontaktraten und Infektionswahrscheinlichkeiten.

2.2.2 Simulation von Netzwerkverkehr

Bei der Simulation von Netzwerkangriffen, wird in der Regel auf einem niedrigen Abstraktionslevel gearbeitet. Dafür wird der Netzwerkverkehr, der eigentlich erzeugt werden würde,

mittels Software (z.B. OMNeT++[2]) simuliert. Es können dabei komplexe Netzwerke mit gängigen Komponenten nachgebildet werden, um zum Beispiel vorteilhaftes Handeln - wie Abschalten von Routern zum Trennen von Netzsegmenten - im Fall von Einbrüchen in Netzwerke bis zu DDoS-Angriffen in Trainingsszenarien zu ermitteln (Kotenko, 2007).

2.2.3 Multiagenten-Simulation

Um die Ausbreitung von Schadsoftware - wie sie bei Cyberangriffen zum Einsatz kommt - vom Angreifer hin zum Ziel zu simulieren, werden auch Multiagentensysteme eingesetzt. Ein existierender Ansatz ist das Modellieren von Agenten, die Schadsoftware repräsentieren. Dabei wird versucht, das Ausbreitungsverhalten von Schadsoftware in die Agenten zu modellieren und das Replizieren auf einem nachgebildeten Netz von Rechnern, auf denen ein verteiltes Multiagentensystem läuft, zu simulieren (Leszczyna u. a., 2008).

2.2.4 Simulation von Benutzerverhalten

Die Simulation von Benutzerverhalten kann im IT-Sicherheitsumfeld unter anderem dafür benutzt werden, Intrusion Dection Systeme (IDS) zu trainieren oder auch zu testen (Garg u. a., 2006). Dabei ist diese Möglichkeit für IDSysteme interessant, die mit Mustererkennung arbeiten, um Eindringungsversuche zu erkennen. Unter anderem ist das Benutzerverhalten eine Quelle für solche Muster. Dabei muss es sich nicht zwingend um einen bewussten Versuch handeln, sondern kann auch unbeabsichtigte gefährliche Interaktionen einschließen. Da das Testen oder Trainieren eines IDS anhand von Nutzerverhalten - sollte es manuell durchgeführt werden - ein aufwändiger Prozess ist, stellt die Simulation dieses Verhaltens einen großen Effizienzgewinn dar (Garg u. a., 2006).

2.2.5 Bewertung

Die vorgestellten Simulationsansätze konzentrieren sich jeweils nur auf eine Schicht. Die Ausnahme stellen die mathematischen Modelle dar, die Faktoren aller Ebenen in den statistischen Werten berücksichtigen können. Durch Verwenden statistischer Werte geschieht dies in einer allgemeinen und abstrakten Form; individuelles Benutzerverhalten zum Beispiel findet keine Berücksichtigung. Mit den mathematischen Modellen lassen sich keine konkreten Empfehlungen herleiten, die eine Eindämmung von Schadsoftware oder Angriffen ermöglicht. Unter Berücksichtigung der in Abschnitt 2.1 erläuterten möglichen Angriffsebenen und deren Abhängigkeiten untereinander,

[2] http://www.omnetpp.org/

lässt sich schlussfolgern, dass bei einer Simulation die Betrachtung mehrere Ebenen sinnvoll ist. Dies geschieht bei den bisherigen Ansätzen nicht.

3 Angestrebtes Ziel

Bei einer Simulation von Cyberangriffen können verschiedene Dinge von Interesse sein. Dabei kommt es darauf an, welches Ziel eine Simulation hat. Sollen beispielsweise Netzwerkadministratoren geschult werden, ist sicher die Simulation eines Netzwerkes mit seinem Nachrichtenverkehr geeignet, um relevante Schlüsse zu ziehen. Ereignisse, wie zum Beispiel der Einsatz von Stuxnet oder Botnetzen, zeigen allerdings, dass die Ausbreitung von Schadsoftware einen interessanten und maßgeblich relevanten Aspekt bei Cyberangriffen darstellen. Durch die zunehmende Vielfalt an mobilen Geräten, ergeben sich neue Angriffsziele und Verbreitungswege für Angriffe. So sind neben E-Mails, Netzwerkfreigaben und anderen schon länger bekannten Möglichkeiten, auch Smartphones, die per USB zum Laden des Akkus an einen PC angeschlossen werden, eine neue Möglichkeit Angriffe zu starten oder vorzubereiten (Zhaohui Wang, 2010). Auch Wechselmedien - wie USB-Sticks - erlangten vor kurzem wieder an Bedeutung, um Angriffe durchzuführen: Ein Verbreitungsweg von Stuxnet sind USB-Wechselmedien (Symantec Corporation, 2011).
Vor diesem Hintergrund soll eine Möglichkeit geschaffen werden, die Ausbreitung von Schadsoftware und ihren Effekt auf befallene Systeme zu simulieren. Dabei soll auch das Anwenderverhalten berücksichtigt werden, da dies maßgeblichen Einfluss auf die Verteilung hat. Zusätzliche Beachtung sollen mobile Geräte finden, die in Verbindung mit dem Nutzerverhalten ebenfalls als Übertragungsweg in Frage kommen.
Anhand der Erkenntnisse einer solchen Simulation soll es möglich sein, präventive Maßnahmen wie Verbessung des Patchlevels von Systemen, spontane Abschottung, besondere Anwenderschulung oder Notwendigkeiten von Strukturänderungen zu erkennen und zu entwickeln.

3.1 Realisierung

Von denen in Absatz 2.2 vorgestellten Möglichkeiten eine Simulation von Cyberangriffen zu realisieren, ist der Ansatz ein Multiagentensystem zu verwenden, um die Vision umzusetzen am sinnvollsten. Ein mathematisches Modell kommt nicht in Frage, da - wie bereits erwähnt - individuelles Nutzerverhalten keine Berücksichtigung findet.
Dabei ist der Ansatz Schadsoftware als Software-Agenten zu modellieren geeignet, da Schadsoftware nach bestimmten Mustern agiert und Ziele verfolgt. Systeme und Peripherie, die von

Schadsoftware befallen werden können und bei ihrer Ausbreitung beteiligt sind, können ebenfalls als Software-Agenten modelliert werden. Bei einer Infektion findet ein Informationsaustausch statt, der durch den Versand von Nachrichten innerhalb eines Multiagentsystems abgebildet werden kann.
Die Modellierung von Anwendern, die wie erwähnt maßgeblich bei der Verbreitung beteiligt sein können, verhält sich ähnlich. Bei der Bedienung von Systemen findet ebenfalls ein Informationsaustausch statt. Zudem ist es möglich, unterschiedliches Nutzerverhalten für verschiedene Agenten zu erstellen. Angedacht ist hier vorerst ein Verhalten, das sich einfach vom technischen Wissensstand eines Agenten ableiten lässt. Technisch versierte Agenten gehen gewissenhafter und bedachter mit Systemen um, als es technische Laien-Agenten machen.
Die Informationen, die über ein System gesammelt werden müssen, um die Ausbreitung von Schadsoftware zu simulieren, erstrecken sich von der genauen Betriebssystemversion über installierte Anwendungen bis hin zu laufenden Diensten und ihre Versionsnummer. Anhand der Versionen lässt sich identifizieren, für welchen Angriff ein System anfällig ist. Als Quelle für Anfälligkeiten können Exploit- und Vulnerability-Datenbanken genutzt werden (z.B. http://www.securityfocus.com oder http://www.exploit-db.com). Um das Nachbilden von Netzen zu vereinfachen und eine manuelle Zusammenstellung von Komponenten möglichst zu vermeiden, sollen hier unterstützende Werkzeuge entwickelt werden, die sich schon existierender Techniken wie Port- oder Vulnerabilityscanner bedienen. Mögliche Programme die hier genutzt werden können sind zum Beispiel Nmap[3] und OpenVAS[4]. Scanergebnisse sollen so zusammengefasst werden, dass sie für den Aufbau des virtuellen Netzes für die Simulation genutzt werden können und diesen unterstützen.

3.2 Risiken

Bei den gesetzten Zielen ergeben sich Risiken zur Realisierung auf verschiedenen Ebenen. Zum einen muss ermittelt werden, wie detailliert der Ausbreitungsvorgang beschrieben werden muss, um hilfreiche Informationen zu gewinnen. Hier stellt sich beispielsweise die Frage, ob es nötig ist, eine Simulation komponenten-, prozess-, dienst- oder systembasiert durchzuführen, um ausreichende Erkenntnisse über Ausbreitungsursachen und ihre Vermeidung zu erlangen.
Zum anderen existiert ein ähnliches Problem beim Modellieren der Verhaltensweise von Schadsoftware. Da sie sich häufig erst in Verbindung mit der Interaktion eines Benutzers ausbreiten kann, ist hier noch unklar, wie man eine möglichst realistische Ausbreitung simuliert.

[3] http://www.nmap.org
[4] http://www.openvas.org

Außerdem fehlen zum Beschreiben von realitätsnahen Szenarien aktuell Informationen zum Aufbau von kritischen Infrastrukturen. Diese in ausreichendem Maße in Erfahrung zu bringen könnte ein Problem darstellen. Sollte dies nicht möglich sein, ließe sich aber auch vorerst mit fiktiven Netzen oder Unternehmensnetzen als Vorbild arbeiten.

4 Zusammenfassung und Ausblick

Im Hinblick auf den weiteren Verlauf des Masterstudiums wurden mit dieser Ausarbeitung die Motivation zur Simulation von Cyberangriffen und Möglichkeiten gezeigt, diese zu realisieren. Für die weitere Arbeit an diesem Thema wurde außerdem ein Ziel festgelegt und ein möglicher Lösungsansatz diskutiert. Aufbauend auf diesen Erkenntnissen muss zunächst eine sinnvolle Granularität des zu realisierenden Systems ermittelt werden.
In Zukunft soll es möglich sein, Szenarien durchzuspielen, die vor allem kritische Infrastrukturen betreffen. Um sinnvolle Testszenarien zu ermitteln, wird dabei auf die Hilfe der Schutzkommission beim Bundesmindesterium des Innern[5] zurück gegriffen.

[5] http://www.schutzkommission.de

Literaturverzeichnis

[Arquilla 2011] Arquilla, John: From blitzkrieg to bitskrieg: the military encounter with computers. In: *Commun. ACM* 54 (2011), Oktober, S. 58–65. – URL http://doi.acm.org/10.1145/2001269.2001287. – ISSN 0001-0782

[Capgemini 2009] Capgemini: IT-Trends 2009 / Capgemini. 2009. – Forschungsbericht

[Erickson 2009] Erickson, John: *Hacking Die Kunst des Exploits*. dpunkt.verlag, 2009

[Garg u. a. 2006] Garg, A. ; Vidyaraman, S. ; Upadhyaya, S. ; Kwiat, K.: USim: a user behavior simulation framework for training and testing IDSes in GUI based systems. In: *Simulation Symposium, 2006. 39th Annual*, april 2006, S. 8 pp.. – ISSN 1080-241X

[John J. Kelly 2008] John J. Kelly, Lauri A.: eWMDs. In: *Policy Review No. 152* (2008). – URL http://www.hoover.org/publications/policy-review/article/5662. – abgerufen 10.01.2012

[Kevin D. Mitnick 2002] Kevin D. Mitnick, Steve W.: *The Art of Deception: Controlling the Human Element of Security*. Wiley Publishing, 2002

[Kotenko 2007] Kotenko, I.: Multi-agent Modelling and Simulation of Cyber-Attacks and Cyber-Defense for Homeland Security. In: *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2007. IDAACS 2007. 4th IEEE Workshop on*, sept. 2007, S. 614 –619

[Leszczyna u. a. 2008] Leszczyna, Rafał ; Fovino, Igor N. ; Masera, Marcelo: MAISim: mobile agent malware simulator. In: *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*. ICST, Brussels, Belgium, Belgium : ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008 (Simutools '08), S. 35:1–35:6. – URL http://dl.acm.org/citation.cfm?id=1416222.1416262. – ISBN 978-963-9799-20-2

[Lora Billings 2002] Lora Billings, Ira B. S.: A unified prediction of computer virus spread in connected networks. In: *Physics Letters A* 297 (2002), S. 261–266

[Messner 2011] Messner, Michael: *Metasploit: Das Handbuch zum Penetration-Testing-Framework*. 1. Auflage. d.punkt Verlag, 2011. – 13–58 S

[Symantec Corporation 2011] Symantec Corporation: W32.Stuxnet Dossier / Symantec Corporation. 2011. – Forschungsbericht. abgerufen 01.04.2011

[Zhaohui Wang 2010] Zhaohui Wang, Angelos S.: Exploiting Smart-Phone USB Connectivity For Fun And Profit. In: *ACSAC '10 Proceedings of the 26th Annual Computer Security Applications Conference*, 2010