Technische Universität Ilmenau 11. April 2024

Vorlesung

Schutz von Kommunikationsinfrastrukturen

Prof. Dr.-Ing. Günter Schäfer

NeXTeX

Vorlesungsmitschrift von
Adrian Schollmeyer

Inhaltsverzeichnis

1 Introduction

1.1 Motivation

Moderne Infrastruktur ist zunehmend abhängig von funktionierenden Kommunikationsnetzen. Verfügbarkeit und Sicherheit beim Betrieb von solchen Kommunikationsnetzen und daran angeschlossenen Teilnehmern wird immer wichtiger. Dabei ist insbesondere nicht nur die Verfügbarkeit wichtig, sondern auch die Sicherheit, da Angriffe auch die Verfügbarkeit und Stabilität des Systems beeinträchtigen können.

1.2 Threats and Security Goals

Eine **Bedrohung** ist ein mögliches Ereignis oder eine Sequenz von Ereignissen, die zu einer Verletzung von mindestens einem Sicherheitsziel führen kann. Im Unterschied zu einem **Angriff**, der tatsächlich passiert, liegt eine Bedrohung bereits vor, wenn es nur möglich ist, einen Angriff zu fahren. Beim Schutz von Kommunikationsinfrastruktur sollte nicht auf Angriffe reagiert, sondern Bedrohungen im Vorfeld vermieden werden.

Sicherheitsziele können abhängig von verschiedenen Faktoren definiert werden, beispielsweise anhand der Anwendungsumgebung:

Telekommunikationsanbieter Privatsphäre der Nutzer, Zugriffsbeschränkung für administrative Funktionen, Ausfallsicherheit

Firmen-/Privatnetze Privatsphäre, Verschluss von Firmengeheimnissen, Authentizität von Nachrichten, Ausfallsicherheit

Alle Netzwerke Schutz vor Eindringlingen von außen

Technisch definiert wird sich auf fünf Ziele beschränkt:

Vertraulichkeit (von Daten, Identitäten, etc.)

Integrität (Sicherstellung, dass Daten auf dem Kommunikationsweg nicht verändert worden sein können; benötigt auch Authentizität der Nachricht)

Accountability (Nachweisbarkeit, welche Entität für welches Ereignis verantwortlich ist)

Kontrollierter Zugriff (Durchsetzung von Regeln, welche Entitäten welche Rechte im System haben)

Verfügbarkeit (Systeme sollten verfügbar sein, korrekt funktionieren und in ihrer Leistungsfähigkeit nicht beeinträchtigt werden)

Angriffe lassen sich auch in Kategorien einteilen:

Masquerade (der Angreifer kann sich als jemand anderes ausgeben)

Eavesdropping (Abhörung von übermittelten Nachrichten)

Authorization Violation (ein Angreifer macht etwas, wofür er eigentlich nicht die nötigen Berechtigungen haben sollte)

Verlust oder Veränderung von Daten

Repudiation (Abstreiten von Kommunikationsvorgängen, erfordert meist anwendungsspezifisches Wissen)

Fälschung von Informationen (Angreifer erzeugt neue Informationen im Namen einer andere Entität)

Sabotage

Verschiedene Angriffe bedrohen Sicherheitsziele in verschiedener Weise. Es ist jedoch immer möglich, dass die Verletzung eines Sicherheitsziels dazu führen kann, dass andere, gefährlichere Angriffe gefahren werden können (bspw. kann mit Eavesdropping ein Root-Passwort ausgelesen werden, wodurch Maskerade möglich ist).

1.3 Security Analysis of Layered Protocol Architectures

Die erste Frage bei der Analyse ist, an welchen Stellen ein Angreifer welche Angriffe fahren kann. Je nach Ort kann ein Angreifer dann an verschiedenen Schichten der Protokollarchitektur angreifen. Je nach Schicht sind wiederum verschieden mächtige Angriffe möglich.

Auf der Nachrichtenebene (sprich, wo einzelne Pakete der eingesetzten Protokolle verschickt werden), lässt sich analysieren, welche Angriffe auf die **PDUs** (Protocol Data Units) gefahren werden können. Für einen erfolgreichen Angriff darf es keine Nebeneffekt auf andere Verbindungen, idealerweise auch nicht auf die betroffene Verbindung geben. Ansonsten besteht das Risiko, dass der Angriff scheitert oder sogar vom Opfer erkannt wird.

1.4 Security Analysis of Communication Infrastructure

Neben den Angriffen auf die Informationsübertragung sind auch Angriffe auf die Systeme, die Teil des Kommunikationsnetzes sind, wichtig, u. A.

- Endsysteme
- Router
- wichtige Infrastrukturdienste (z. B. DNS, E-Mail, Webserver, etc.)

Dabei erweitert sich die Analyse deutlich und wird viel anwendungs- und systemspezifischer.

1.5 Towards Systematic Threat Analysis

Die Erstellung beliebiger Listen möglicher Angriffe ist keine besonders zielführende Methode. Dabei ist es nur schwer möglich, die Vollständigkeit der identifizierten Angriffe zu zeigen.

Ein **Bedrohungsbaum** beschreibt Bedorhungen auf verschiedenen Eben mit verschiedenen Leveln an Abstraktion. An den obersten Knoten werden die generellen Ziele definiert, während beim Absteigen in den Baum immer konkretere Angriffsszenarien und Bedrohungen notiert werden. Irgendwann erhält man Blattknoten, die sehr detaillierte Bedrohungen beschreiben und die in einer (weniger beliebige) Bedrohungsliste aufgenommen werden können.

Bei der Erstellung von Bedrohungsbäumen sollte darauf geachtet werden, auf jeder Analyse vollständig zu arbeiten, bspw. in dem binäre Entscheidungen in Knoten repräsentiert werden (sodass alle möglichen Fälle abgedeckt werden). Weiterhin können Knoten auch in verschiedenen logischen Beziehungen (und, oder) stehen, je nachdem ob ein Angreifer bspw. nur eine von mehreren oder auch mehrere Hürden auf einmal überwinden können muss, um einen Angriff durchzuführen (vergleiche: Stellen, um in ein Haus einzubrechen vs. Sicherheitsmaßnahmen wie Zäune, Wachhunde, etc.). Sind nun Angriffsszenarien bekannt, kann der Aufwand (die Kosten), Angriffe durchzuführen, abgeschätzt werden, um zu analysieren, welche Angriffe am wahrscheinlichsten sind. Für oder-verknüpfte Knoten muss dabei das Minimum genommen werden (angenommen der Angreifer kennt den Aufwand, wird er den geringsten Aufwand wählen), während bei und-verknüpften Knoten das Maximum gewählt werden muss (er muss mindestens diesen Aufwand betreiben, um den Angriff durchzuführen).

Daneben stellt sich auch die Frage, wie viele Personen zu einem Bestimmten Angriff bereit sind. Das ist vor allem eine Frage dessen, was geschützt werden soll (vgl. Staatsgeheimnisse vs. eine Privatwohnung) und wer als Angreifer infrage kommt (und welche Möglichkeiten die Angreifer dann potentiell haben). Dies lässt sich im Bedrohungsbaum nicht direkt modellieren. Stattdessen muss die resultierende Angriffsliste auf Kosten und Gewinn analysiert werden und die Frage ist, welche Angreifer dann noch zu welchen Angriffen bereit sind.

1.6 System Security Engineering Process

Dies erlaubt ein etwas systematischeres Vorgehen:

- Spezifikation der Systemarchitektur
- Identifikation von Bedrohungen, Schwachstellen und Angriffstechniken
- Abschätzung von Risiken (zusätzliche Attribute am Bedrohungsbaum)
- Priorisierung von Schwachstellen
- Identifikation und Installation von Sicherungsmaßnahmen (für Schwachstellen mit hoher Priorität)

- Neuanalyse/Iteration der o. g. Schritte

Es kann dabei prinzipiell auch möglich sein, dass der Angreifer damit rechnet, dass bestimmte Sicherungsmaßnahmen ergriffen werden. Es besteht also grundsätzlich die Gefahr, dass durch die Installation von solche Maßnahmen auch neue Bedrohungen entstehen, deren Ausnutzung potentiell schon vom Angreifer geplant sind. Es kann also auch Sicherungsmaßnahmen geben, die tatsächlich kontraproduktiv sind.

1.7 A High Level Model for Internet-based IT-Infrastructure

Die Hauptunterscheidung ist zwischen private Netzen (d. h. Zeug, was keine Dienste anbietet, sondern mit Diensten spricht; bspw. Heimnetze, Sensornetze, etc.), Support-Infrastruktur (Transportnetze wie das Internet) und ISP-Netzwerke (Bereitstellung von Diensten, Cloud-Hosts, Rechenzentren, Mobilkommunikationsnetze).

Grundsätzlich dürfen physikalische Bedrohungen (bspw. Zerstörung eines Rechenzentrums) nicht missachtet werden und physische Redundanz ist immer wichtig, um Verfügbarkeit zu garantieren. Dennoch sind solche Bedrohungen für diese Vorlesung weniger relevant und werden nicht wirklich behandelt.

Link-basierte Bedrohungen auf physischer Ebene sind zwar theoretisch möglich, lohnen sich aber meist weniger als Angriffe auf dem Data Link Layer. Die Untersuchung und Behandlung solcher Bedrohungen ist Teil der Veranstaltung „Network Security“.

Beim Network Layer werden hier auch Dienste hinzugezogen, die nicht direkt auf dem Network Layer laufen, aber für den Betrieb des Netzes an sich notwendig sind (bspw. DNS). Die Anwendungsschicht wird in diesem Bedrohungsbaum vereinfacht und ist nicht vollständig. Dieser Bereich ist auch sehr stark davon abhängig, welche Anwendungen im Netz betrieben werden.

1.8 Countering Attacks

1.8.1 Principle Classes of Action

Prävention Umfasst alle Maßnahmen, um Angreifer davon abzuhalten, erfolgreiche Angriffe durchzuführen (bspw. Verschlüsselung, Signaturen, Firewalls) Dies muss grundsätzlich passieren, *bevor* der Angriff stattfindet.

Erkennung Umfasst alle Maßnahmen, um laufende oder vergangene Angriffe zu erkennen. Dazu gehören Audit Trails, Traffic Monitoring (idealerweise on-the-fly), etc. Wird ein laufender Angriff erkannt, sollten Maßnahmen ergriffen werden, um den Angriff entweder zu unterbinden oder zunächst zu entscheiden, ob der Angriff zu echten Schäden führen oder der Angreifer ermittelt werden kann.

Eine Variante der Erkennung von potentiellen Angreifern ist der Betrieb eines **Honeypots**. Diese Systeme sind am Netz erreichbar mit dem Ziel, dass Angreifer diese angreifen. Auf solchen Systemen sollten keine echten Dienste laufen, sodass der Zugriff auf das System direkt als Angriff erkannt werden kann. Wichtig ist

hierbei jedoch auch wieder, dass der Honeypot selbst kein Einfallstor in echte Kommunikationsinfrastruktur darstellt.

Reaktion Umfasst alle Maßnahmen, die in Reaktion auf vergangene oder laufende Angriffe unternommen wurden.

1.8.2 Safeguards Against Information Security

- Physical Security (Beschränkung von physischem Zugang zu Servern u. Ä.)
- Personnel Security (Überprüfung von Personal, welches Zugriff zu Servern u. Ä. hat)
- Administrative Security (Ausbildung von Personal, Einbringung von Fremdsoftware, Prozeduren und Workflows zur Untersuchung von Sicherheitsvorfällen, Review von Audit Trails etc.)
- Emanations Security (Abstrahlung von Geräten, etc.)
- Media Security (Kontrolle darüber, wie sensible Informationen (sicher) reproduziert und zerstört werden können, Sicherhung der Speichermedien, Virenscans, etc.)
- Lifecycle Controls (Kontrolle des Softwareentwicklungsprozesses, Programmierstandards, etc.)
- Computer/System Security (Schutz der verarbeiteten Informationen und verarbeitenden Geräte)
- Kommunikationssicherheit (Schutz der Informationen auf dem Transportweg, Schutz der Kommunikationsinfrastruktur)

1.9 Terminology

Security Service Abstrakter Dienst, welcher eine bestimmte Sicherheitseigenschaft garantieren soll. Kann mithilfe von Kryptographie, Protokollen, aber auch konventionellen Mitteln (z. B. Ablegen des Datenträgers in einem Tresor) garantiert werden. Meist werden mehrere Dienste miteinander kombiniert.

Cryptographic Algorithm

cryptographic algorithm

Cryptographic Protocol

cryptographic protocol Regeln, wer wann welche Berechnungen durchführen und welche Nachrichten verschiken muss, um bestimmte Sicherheitsziele zu erreichen (z. B. für Authentisierung, Schlüsselaustausch, etc.).

1.10 Security Services

Authentisierung

authentication

Integrität

integrity

Vertraulichkeit

confidentiality

Zugriffskontrolle

access control

Nichtabstreitbarkeit

non-repudiability Es soll sichergestellt werden, dass kein Kommunikationspartner später abstreiten kann, an einer Kommunikation teilgenommen zu haben. Dies lässt sich eigentlich nur durch Zertifizierung durch eine unabhängige dritte Stelle sicherstellen.

2 Security Aware System Design and Implementation

2.1 Problems of Practical System Security

Es ist unmöglich, die Sicherheit eines moderat komplexen Systems zu zeigen. Allgemein ist es schwierig, die Abwesenheit einer Eigenschaft (hier: Bedrohung) zu zeigen. Daher ist es wichtig, möglichst einfache Systeme einzusetzen, wenn Sicherheit garantiert werden soll. Zusätzlich müssen die Quellen beherrscht werden, von denen die Systeme kommen. Meist ist dabei Software das Problem, die durch schlechtes Softwaredesign oder schlechte Implementierung Sicherheitsprobleme haben. Systemadministratoren müssen dann häufig Sicherheitspatches einpflegen, was zu zusätzlicher Last (und potentieller Nachlässigkeit) bei der Wartung führt, wodurch wieder Fehler gemacht werden können.

Weitere Probleme sind der Einsatz von Low-Level-Programmiersprachen, die einfache Angriffe ermöglichen (bspw. Buffer Overflow), schlechte Standardkonfigurationen von Software, Erweiterbarkeit von Software durch Updates und dynamisch ladbare Erweiterungen (e. g. Plugins) und der Mangel an Diversität in beliebten Rechenumgebungen (bspw. Windows auf PCs, Linux auf Servern, Cisco IOS für Router, etc.). Durch die immer steigende Geschwindigkeit bei der Softwareentwicklung kommt es zu immer kürzeren Entwicklungszyklen mit schlechter Spezifikation von Anforderugen, sofern diese überhaupt existieren. Gerade letztes führt dazu, dass Sicherheitsaspekte nicht oder nur unzureichend beachtet werden.

Stichwortverzeichnis