Informationssicherheit und IT-Forensik

1-06 – Sicherheitsmodelle und Zugriffskontrolle

Grundlagen

Lernziele dieser Einheit

- Grundbegriffe, deren Bedeutung und Zusammenhang verstehen und anwenden können
- Die Grundprinzipien verbreiteter Zugriffskontroll-Modelle und Zugriffskonzepte verstehen und als Modell für das Design eines Betrieblichen Anwendungssystems anwenden können
- Das Konzept von verschiedenen Implementierungen in Windows und Linux verstehen

Grundlagen

Sinnvolles Grundprinzip: need to know

- Need to know = nur das wissen (können), was man für eine konkrete Aufgabe auch wirklich wissen muss
- Minimalprinzip

- Quelle: https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/m/m02/m02008.html

- Bezogen auf Zugriffsmodelle
  - Weitere Einschränkung, die auch dann gilt, wenn eine Person eigentlich grundsätzlich/generell Zugriff z.B. auf Dokumente einer Sicherheitsstufe hat („Top secret“) -> Zugriff wird dann dennoch nur auf Informationen eines bestimmten Projektes erlaubt

Grundlagen

Objekte und deren Relationen

- Objekt
  - Zu schützende Einheit
  - Granularität muss festgelegt werden (Bildung von Objekten, die feiner granulare Objekte umfassen -> diese werden nicht mehr einzeln betrachtet)
  - Z.B. eine Datei in einem Dateisystem, ein PC oder eine Anwendung
- Subjekt
  - Will auf ein Objekt zugreifen, bzw. tut dies tatsächlich
  - Granularität muss festgelegt werden (Bildung von Objekten, die feiner granulare Objekte umfassen -> diese werden nicht mehr einzeln betrachtet)
  - Z.B. ein Benutzer oder eine Benutzergruppe
- Quelle: Claudia Eckert, IT-Sicherheit, 9. Auflage

Grundlagen

Objekte und deren Relationen

- Zugriffsrechte
  - Universelle Rechte (z.B. read/write auf eine Datei für eine Benutzergruppe)
  - Objektspezifische Rechte (z.B. INSERT oder UPDATE in SQL)
- Zugriffsbeschränkungen
  - Einfach (Recht wird einem Subjekt pauschal eingeräumt), z.B. Leserecht auf eine Datei
  - Komplex (Recht wird durch weitere Bedingungen eingeschränkt eingeräumt), z.B. Leserecht auf eine Datei wird nur 1x gewährt
- Quelle: Claudia Eckert, IT-Sicherheit, 9. Auflage

Grundlagen

Zugriffskontrolle

- WER darf WAS auf/mit welcher RESSOURCE
- Wird kontrolliert durch BERECHTIGUNGEN

Grundlagen

Zugriffskontrolle

- Kann auf unterschiedlichen Ebenen erfolgen (Schichtenmodell)
  - Hardware
  - OS
  - Middleware
  - Anwendungen
- Herausforderung: vertikale Abhängigkeiten, z.B. Gefahr von Bypass durch Direktzugriff auf untere Ebenen

Sicherheitsstrategien

Strategische Überlegungen: Zugriffskontroll-Strategien

- Wer ein Betriebliches Informationssystem entwickelt wird ein System zum Management und der Kontrolle von Zugriffsberechtigungen entwickeln (müssen)
- Je nach Schutzanforderungen, Komplexität und insbesondere geplanten Teilprozessen (Zugriffsmöglichkeiten) kommen verschiedene Zugriffskontroll-Strategien zum Einsatz
  - Discretionary Access Control (DAC)
    - Ein Objekteigentümer setzt pro Objekt Zugriffsrechte (benutzerbestimmbar)
  - Mandatory Access Control (MAC)
    - Rechte werden regelbasiert vom System global und mit Vorrang vor benutzerbestimmbaren Rechten gesetzt (z.B. Vertraulich/Geheim/Streng geheim) (systembestimmt)
  - Role-Based-Access-Control (RBAC)
    - Rechte werden einer Rolle zugeordnet. Subjekte werden anschließend je nach Bedarf Rollen zugeordnet (rollenbasiert)
- Quelle: Claudia Eckert, IT-Sicherheit, 9. Auflage

Sicherheitsstrategien

Strategische Überlegungen: Informationsfluss-Strategien

- Informationsfluss-Strategien regeln darüber hinaus, welche Informationskanäle zwischen Subjekten zulässig sind
- Beispiel: Objekte (spezifische Informationen) werden einer Sicherheitsklasse zugeordnet. Informationen dürfen nicht von einer höheren in eine niedrigere Sicherheitsklasse fließen.

Sicherheitsstrategien

Zugriffskontroll-Modelle: Matrix-Modell

- Spalten: Objekte (Datei, Prozess, Socket, …)
- Zeilen: Subjekte (Benutzer, Prozess, PC, …)
- Zellen: Rechte (write, read, control, …)
- Kann statisch oder dynamisch erfolgen (nach Zeit t)
- Quelle: *Graham, G. Scott and Peter J. Denning. “Protection: principles and practice.” AFIPS Spring Joint Computing Conference (1971)*
- Problem: häufig “dünn” besetzt

Sicherheitsstrategien

Zugriffskontroll-Modelle: Zugriffskontroll-Listen (Access Control List, ACL)

- Pro Objekt wird eine Liste verwaltet:
  - WER darf WAS
- Spaltenweise Abbildung einer Matrix
- Hat auch Vorteile und Nachteile…

Sicherheitsstrategien

Zugriffskontroll-Modelle: Role-Based Access Control (RBAC)

- Aufgabenbasiertes Modell
  Rolle = Aufgabe
- Rollen
- Berechtigungen/Zugriffsrechte
- Subjekte/Benutzer
- Sitzungen (aktive Rollen)
- Quelle: *Ferraiolo und Kuhn, Role-Based Access Controls, 15th National Computer Security Conference (1992)*

Zugriffskonzepte

Modell für Mandatory Access Control: Bell-LaPadula

- *„gilt als das erste vollständig formalisierte Sicherheitsmodell“* (Claudia Eckert, IT-Sicherheit, 9. Auflage, S. 281)
- Militärischer Ursprung
- Horizontal dimensionierte Schutzstufen:
  - Streng geheim
  - Geheim
  - Vertraulich
  - Unklassifiziert
- Es gibt klassifizierte Objekte und Subjekte (Sicherheitskategorien)

- Quelle: D. Elliott Bell, Leonard J. LaPadula: Secure Computer Systems: Mathematical Foundations. MITRE Corporation, 1973

Zugriffskonzepte

Modell für Mandatory Access Control: Bell-LaPadula

- *„no-read-up“*
- „no-write-down“
- Write-up ist jedoch zulässig
- Zusätzlich: Zugriffsmatrix für Objekte x Subjekte
- Innerhalb einer Stufe ist DAC möglich

Zugriffskonzepte

Chinese-Wall-Modell

- *“The Chinese Wall policy combines commercial discretion with legally enforcable mandatory controls. It is required in the operation of many financial services organizations and is, therefore, perhaps as significant to the financial world as Bell-LaPadula's policies are to the military.”*
- Quelle: Brewer und Nash, The Chinese Wall Security Policy, 1989 IEEE Symposium on Security and Privacy, S. 206-214

- Ziel: mit Hilfe eines Betrieblichen Anwendungssystems verhindern, dass jemand in einer Organisation Insiderinformation erhält und z.B. für andere Kunden verwenden kann
- Zukünftige Zugriffsrechte eines Subjekts auf Objekte sollen daher von vergangenen abhängen
- Zugriffsmatrix-Modell

Zugriffskonzepte

Chinese-Wall-Modell: Baumorganisation der Objekte und Einteilung in Schutzklassen

Kommunikationsprinzipen

Unsicherer Kommunikations-Default

- „Schatz, ich melde mich, wenn mir unterwegs etwas passiert ist.“
  - Nicht Fail-Safe
- „Schatz, ich melde mich, wenn ich gut angekommen bin.“
  - Fail-Safe
- Grundprinzip z.B. bei Haltesignalen im Bahnbetrieb

Kommunikationsprinzipen

Warrant Canary

- Umkehr eines Prinzips
  - Intuitiv: ich sage dir, wenn etwas nicht stimmt
  - Warrant Canary: ich höre auf, mich zu melden, wenn etwas nicht stimmt
  - Juristisch kontrovers diskutiert
  - Grundsätzliche Gefahr: ein Dritter erlangt die Kontrolle über das System und erzeugt Fake-Canarys
  - Quelle: https://www.reddit.com/r/reddit.com/comments/6ecu/an_isp_that_protects_your_data_from_the_nsa/ (2006)
  - Quelle: https://en.wikipedia.org/wiki/Warrant_canary

Praktische Implementierungen

Einfache Zugriffsberechtigungen in unixoiden Systemen (z.B. Linux)

```
user@system:~$ ls -la
drwxr-xr-x 31 user user  4096 Jan 29 12:20 .
drwxr-xr-x  3 root root  4096 Jul 26  2016 ..
-rw-------  1 user user  5294 Jan 16 16:00 .bash_history
-rw-r--r--  1 user user   220 Jul 26  2016 .bash_logout
-rw-r--r--  1 user user  3771 Jul 26  2016 .bashrc
drwxr-xr-x  2 user user  4096 Okt 24 10:11 Bilder
drwx------ 16 user user  4096 Jan 29 12:20 .cache
drwx------ 18 user user  4096 Jan 20 01:16 .config
-rw-r--r--  1 user user    26 Jul 26  2016 .dmrc
drwxr-xr-x  2 user user  4096 Jul 26  2016 Dokumente
drwxr-xr-x  2 user user  4096 Mai 16  2018 Downloads
drwx------  2 user user  4096 Jan 29 12:21 .gconf
-rw-r-----  1 user user     0 Dez 10 21:29 .gksu.lock
drwx------  3 user user  4096 Jan 29 12:20 .gnupg
drwx------  2 user user  4096 Sep 25  2016 .gphoto
-rw-------  1 user user 10680 Jan 29 12:20 .ICEauthority
drwxrwxr-x  2 user user  4096 Mai 15  2018 iso
drwxr-xr-x  3 user user  4096 Jul 26  2016 .local
drwxrwxr-x 14 user user  4096 Feb 27  2018 Projekt1
drwxrwxr-x 24 user user  4096 Aug  7  2017 Projekt2TopSecret
```

```
user@system:~$ sudo ls -la /
[sudo] Passwort für user: *******
drwxr-xr-x  24 root root  4096 Jan 14 21:45 .
drwxr-xr-x  24 root root  4096 Jan 14 21:45 ..
drwxr-xr-x   2 root root 12288 Jan 14 21:44 bin
drwxr-xr-x   4 root root  3072 Jan 14 21:52 boot
drwxr-xr-x  20 root root  4360 Jan 29 12:20 dev
drwxr-xr-x 145 root root 12288 Jan 14 21:49 etc
drwxr-xr-x   3 root root  4096 Jul 26  2016 home
drwxr-xr-x  25 root root  4096 Jan 16  2018 lib
drwxr-xr-x   2 root root  4096 Feb  8  2018 lib64
drwx------   2 root root 16384 Jul 26  2016 lost+found
drwxr-xr-x   4 root root  4096 Feb 18  2017 media
drwxr-xr-x   2 root root  4096 Apr 21  2016 mnt
drwxr-xr-x   2 root root  4096 Apr 21  2016 opt
dr-xr-xr-x 260 root root     0 Jan 29 12:19 proc
drwx------   8 root root  4096 Jul 16  2018 root
```

Praktische Implementierungen

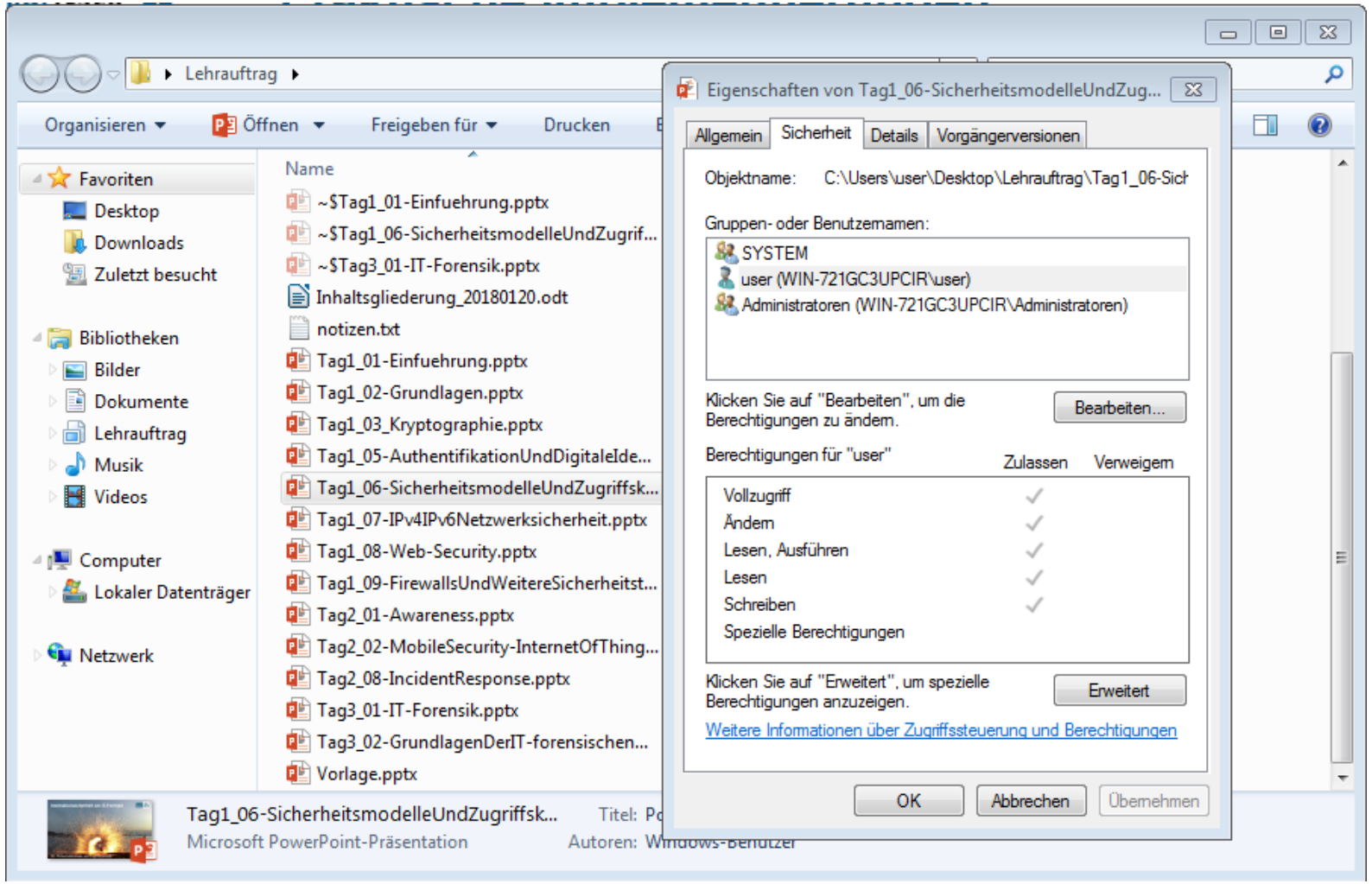
su/sudo/suid in unixoiden Systemen (z.B. Linux)

- su und sudo
- Programme, die in unixoiden Systemen angeboten werden und ermöglichen, Rechte anderer Benutzer zu erlangen, bzw. Prozesse entsprechend zu starten
- Kann Prozess-/Befehls-genau definiert werden
- Beispiele:
  - Normaler Benutzer A wird root
  - Benutzer A wird Benutzer B
  - Root wird normaler Benutzer A
- Vorteil: Passwort des anderen Benutzers muss nicht bekannt sein
- Ist als Liste von Rechten (wer|UID|was) realisiert

- Set User ID (setuid)
- Erweitertes Dateirecht in unixoiden Systemen
- Ist das Zugriffsbit im Dateissytem für ein binary gesetzt, dann wird dies immer auch mit den Rechten des Besitzers ausgeführt
- Beispiel:
  - Login-Prompt, der Zugriff auf die Benutzerdatenbank benötigt
- Vorteil:
  - Einfach, effektiv
- Frage: welche Nachteile erkennen Sie?

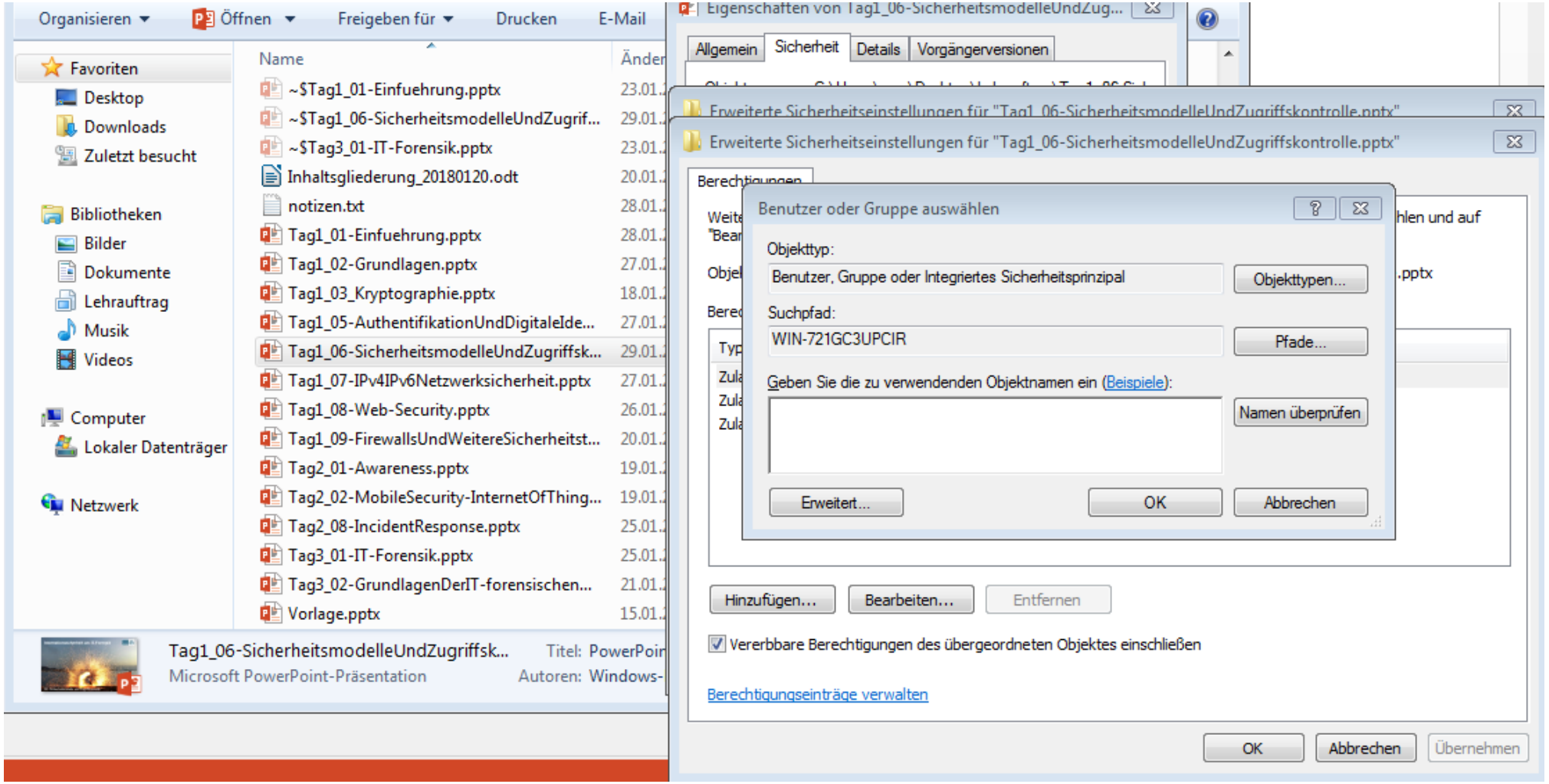
Praktische Implementierungen

ACL in Windows-Dateisystem NTFS



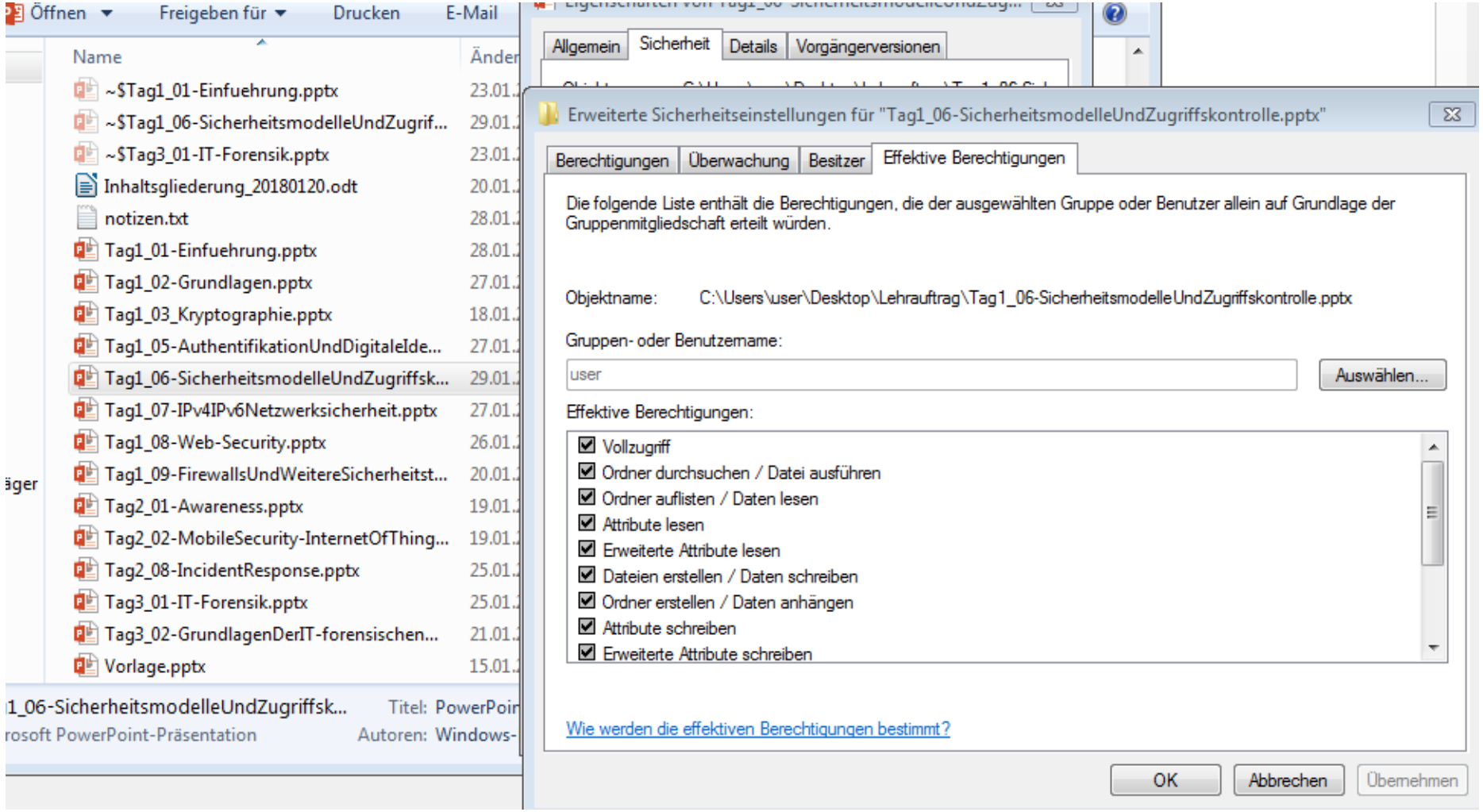
Praktische Implementierungen

ACL in Windows-Dateisystem NTFS



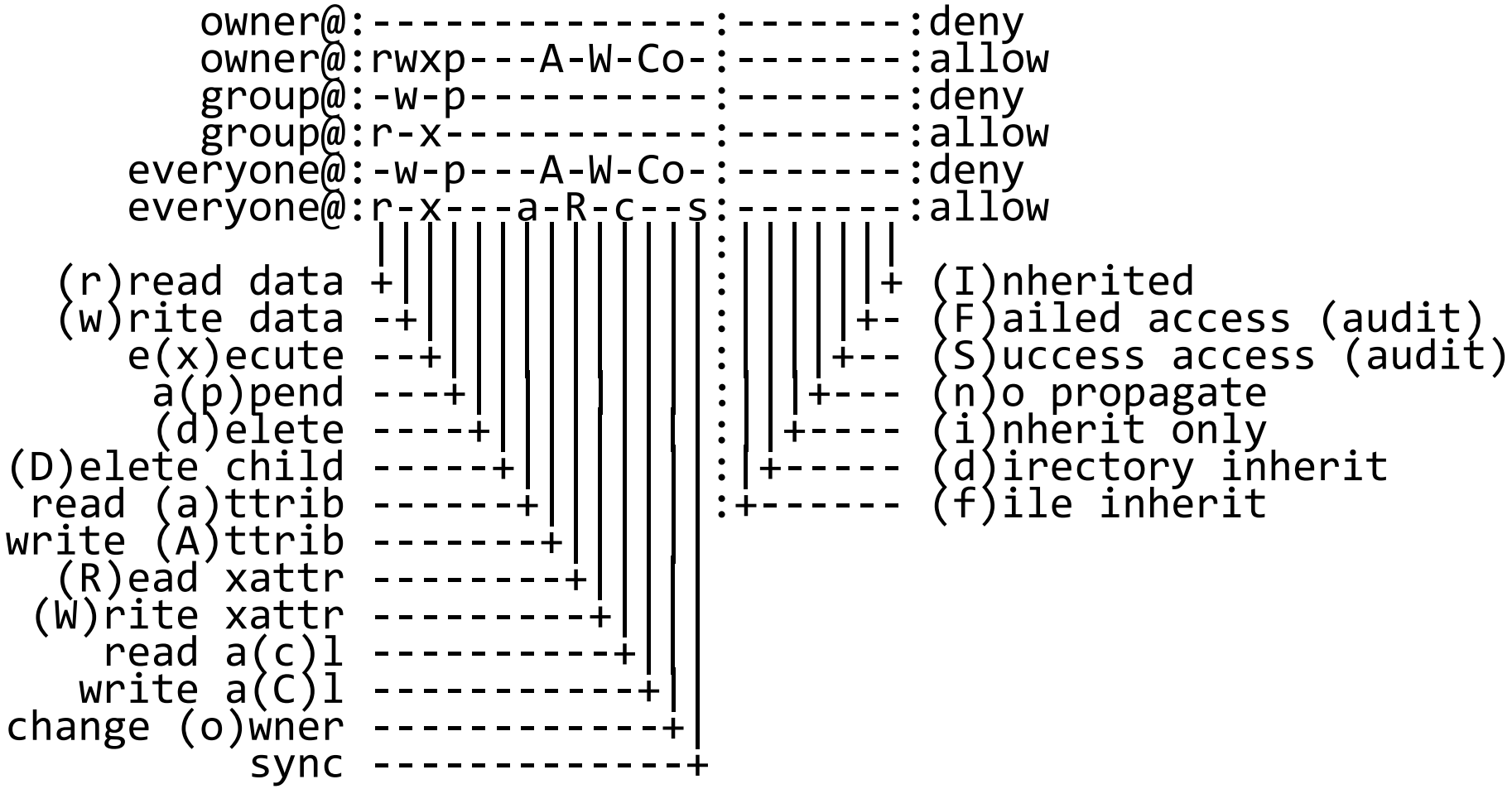
Praktische Implementierungen

ACL in Windows-Dateisystem NTFS



Praktische Implementierungen

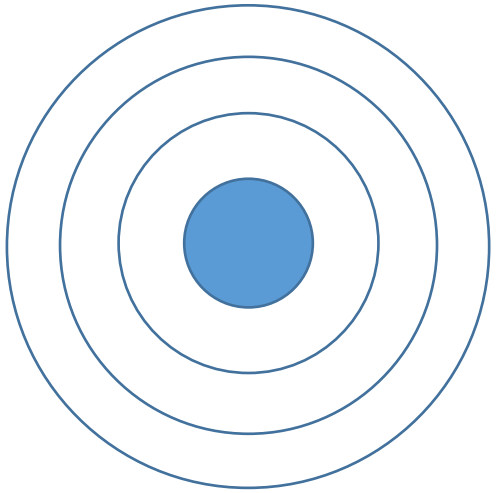
ACL in unixoiden Dateisystemem am Beispiel von ZFS



Praktische Implementierungen

Speicherschutz

- Ursprünglich (8086-Architektur)
  - Real-Mode
  - Direkter Zugriff für alle Programme auf alle Speicherbereiche
- Ab 80286
  - Protected Mode
  - Kein Direkter Zugriff mehr
  - Virtuelle Speicherverwaltung
- X86-Architektur nutzt 2 Ringe: Kernel Mode und User Mode



Praktische Implementierungen

Single Sign On am Beispiel einer Windows Active Directory-Umgebung

- Windows-AD-Umgebungen unterstützen seit Windows 2000 Kerberos
- Bietet Authentifizierung an Systemen sowie an Diensten
- Ticket-Basiert
- Vorteil z.B.: Rechte können zentral verwaltet und vergeben werden
- Nachteil z.B.: Tickets haben eine gewisse Lebenszeit…
  - Silver Ticket
  - Golden Ticket
  - Siehe z.B. https://digital-forensics.sans.org/blog/2014/11/24/kerberos-in-the-crosshairs-golden-tickets-silver-tickets-mitm-more

Zugriffskontrolle in Windows-Systemen

Rechtemanagement in Windows-Clients

- Windows 2000/XP
  - Standalone-Clients: Alle Benutzer sind per Default Admin und dürfen alles
  - Gemanagte Unternehmensumgebungen: normale Benutzer vs. Admins
- Windows Vista/7/…
  - Standalone-Clients: User Access Control
  - Gemanagte Unternehmensumgebungen: normale Benutzer vs. Admins

- Aber: auch in ungehärteten Windows-10-Systemen hat eine Software mit User-Rechten Zugriff auf den Tastatur-Buffer, auch wenn die Anwendung nicht den Input-Fokus hat -> Keylogger

Zugriffskontrolle in Windows-Systemen

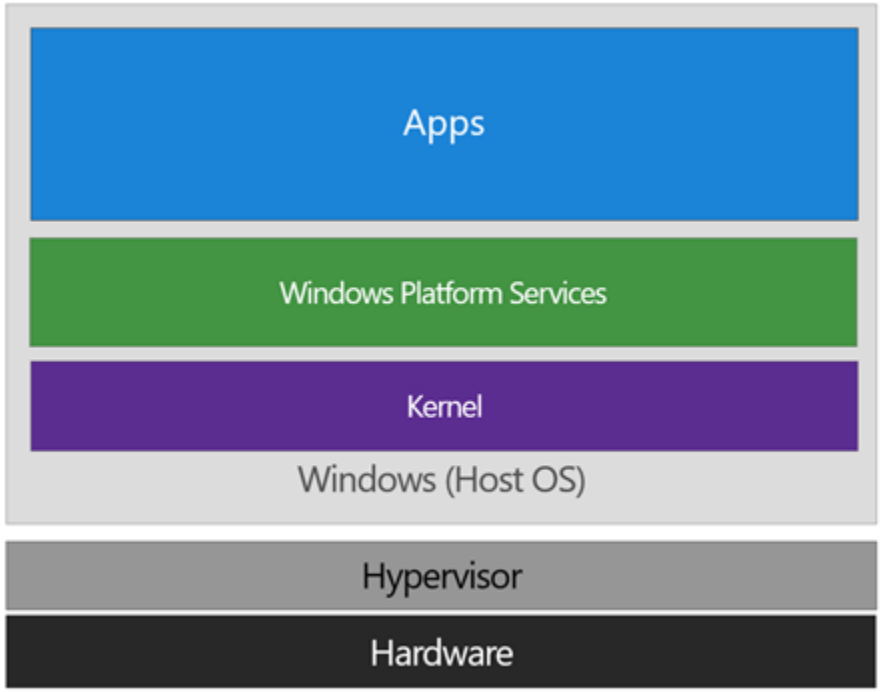
Virtual Secure Mode - Virtualization Based Security (VBS)

- Verfügbar seit Windows 10
- Verwendet Hyper-V und trennt damit eine sichere Zone (Subsystem) vom Hauptsystem ab
- Dort läuft dann z.B. der Local Security Authority Subsystem Service (LSASS) und verwaltet die Passwort-Hashes (Credential Guard)
- Bietet auch die Möglichkeit, nur noch signierte Anwendungen ausführen zu können (Device Guard mit Code Integrity Services)
- Quelle: https://www.golem.de/news/sicherheit-windows-10-und-das-ende-von-malware-1512-117849-2.html

Zugriffskontrolle in Windows-Systemen

Virtual Secure Mode - Virtualization Based Security (VBS)

- Das Betriebssystem läuft nicht mehr direkt auf der Hardware, sondern oberhalb eines Hypervisors

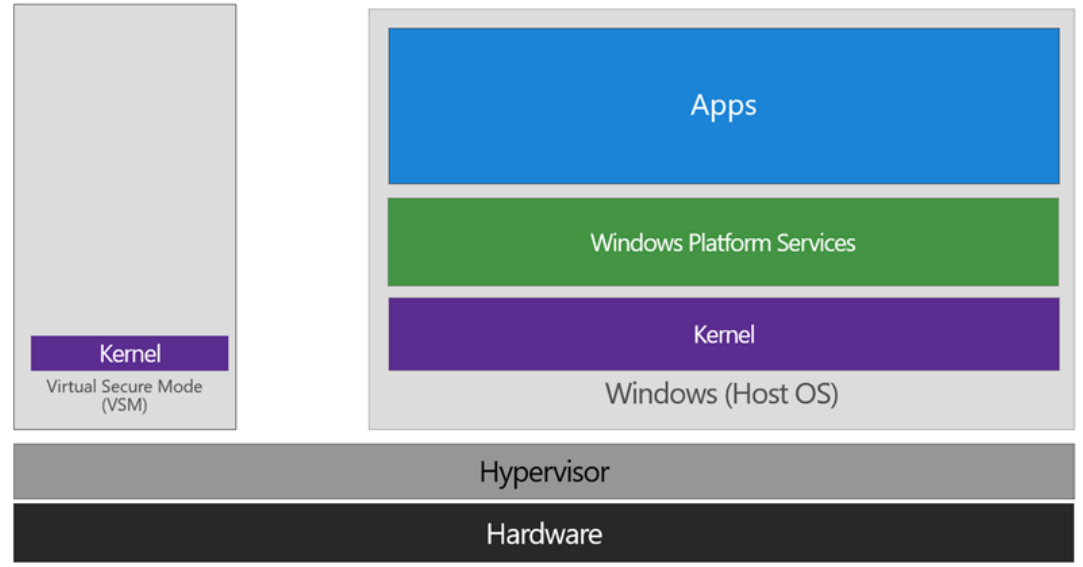


Quelle: https://blogs.technet.microsoft.com/ash/2016/03/02/windows-10-device-guard-and-credential-guard-demystified/

Zugriffskontrolle in Windows-Systemen

Virtual Secure Mode - Virtualization Based Security (VBS)

- Ein Subsystem mit eigenem Kernel läuft nun parallel zum Windows-Host
- Kommunikation erfolgt über festgelegte Wege (API)
- Direkter Zugriff z.B. auf den gesamten RAM von LSASS ist nun nicht mehr möglich
  - LSAIso (LSA Isolated) im VSM und LSASS „Stub“ im Host
- Auch nicht mit SYSTEM-Rechten

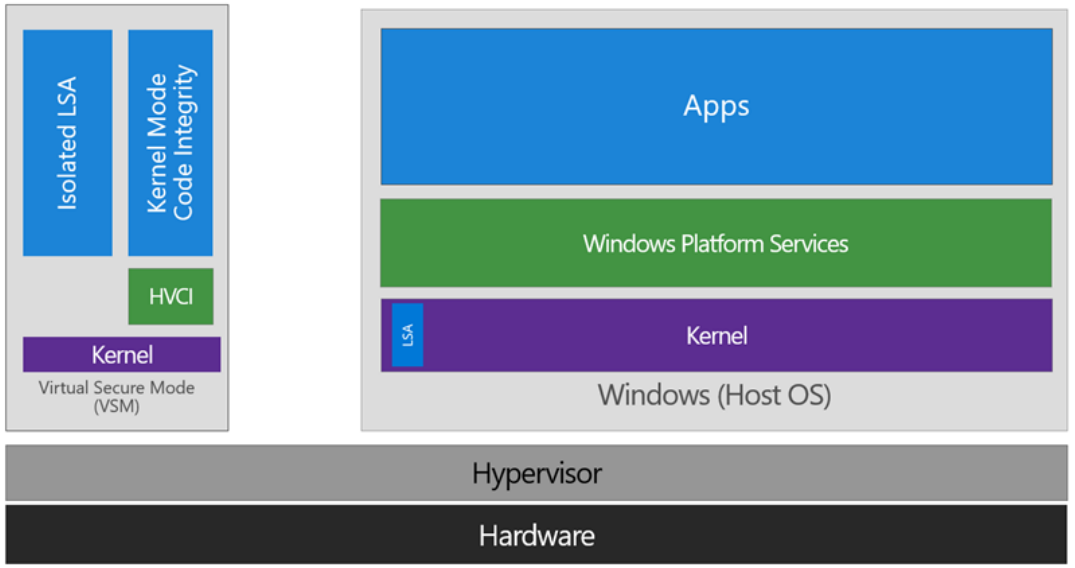


Quelle: https://blogs.technet.microsoft.com/ash/2016/03/02/windows-10-device-guard-and-credential-guard-demystified/

Zugriffskontrolle in Windows-Systemen

Virtual Secure Mode - Virtualization Based Security (VBS)

- Diese Technik ermöglicht auch die geschützte (separierte) Überwachung des ausgeführten Codes im Host



Quelle: https://blogs.technet.microsoft.com/ash/2016/03/02/windows-10-device-guard-and-credential-guard-demystified/

Ein Auftrag an Sie: unser Musterfall

Wiederkehrende, vorlesungs- und übungsbegleitende Übungsaufgabe

- Überarbeiten Sie (als Hausaufgabe) Ihre Planung für die Errichtung einer „smarten“ Einbruchmeldeanlage 2.0 für das Handwerksunternehmen Ihrer Eltern
- Beantworten und begründen Sie unter eigenen Annahmen z.B. folgende Fragen:
  - Ändern Sie Ihre bisherige Planung?
  - Erarbeiten Sie ein Rechtekonzept. Welche Benutzergruppen sollen welche Zugriffsrechte erhalten?
  - Skizzieren Sie mögliche Problemstellen. Wo könnten Probleme lauern?

Live-Hack

Ein unsicheres SUID-Binary nutzen, um auf einem Linux-System Root zu werden

- Z.B. nmap Version 2.02 bis 5.21
- Beispiel für unbedachte Grundkonfiguration / unbedachte Installationen
- https://pentestlab.blog/2017/09/25/suid-executables/
- ACHTUNG:
  - Verwendung auf fremden Systemen ist grundsätzlich illegal