

DATENSCHUTZ *individuell*

Sisyphus im Informationszeitalter aktuell sicher oder stabil unsicher

Software aktuell halten – aber wie?

Olav Seyfarth

Berater für Datenschutz und Informationssicherheit

Vortrag im Rahmen der Arbeitsgruppe „IT-Sicherheit“ des medien forum freiburg

Anlass für den Vortrag war die Beobachtung, dass selbst sicherheitskritische Aktualisierungen selten zeitnah eingespielt werden – und dies in Firmen und Privathaushalten gleichermaßen.

Siehe hierzu heise Security News vom 03.02.10: <http://heise.de/-921139>

Hauptziel des Vortrags ist, auf die technischen Gefahren beim Einsatz angreifbarer Software hinzuweisen. Er gibt IT-Budgetverantwortlichen und Administratoren einen systematischen Überblick über Herausforderungen und Lösungen.

© Olav Seyfarth – 15.04.2010 – alle Marken gehören ihren Eigentümern



Olav Seyfarth

Dipl.-Inform. (FH)

Telefon 07641 / 9671214

Handy 0176 / 62696431

Email olav.seyfarth@datenschutz-individuell.de

WWW <http://www.datenschutz-individuell.de/>

DATENSCHUTZ
individuell

Programm, Prozess, Update, Patch

Process	PID	CPU	Description	Company Name
simpress.exe	204		OpenOffice.org Impress	OpenOffice.org
soffice.exe	212		OpenOffice.org 3.2	OpenOffice.org
soffice.bin	220		OpenOffice.org 3.2	OpenOffice.org

Name	Description	Company Name	Version
{AFBF9F1A-8EE8-4...			
ADVAPI32.dll	Erweiterte Windows 32 Base-API	Microsoft Corporation	6.1.7600.16385
aggmi.dll		OpenOffice.org	3.2.9426.500
animcore.dll		OpenOffice.org	3.2.9426.500
apphelo.dll	Clientbibliothek für Anwendungsk...	Microsoft Corporation	6.1.7600.16481
avgrstr.dll	AVG Resident Shield Starter	AVG Technologies CZ, s.r...	9.0.0.782
avmediami.dll		OpenOffice.org	3.2.9472.500
basegfxm.dll		OpenOffice.org	3.2.9450.500
behelper.uno.dll		OpenOffice.org	3.2.9472.500
bootstrap.uno.dll		OpenOffice.org	3.2.9432.500
canvastoolmi.dll		OpenOffice.org	3.2.9472.500
CFGMGR32.dll	Configuration Manager DLL	Microsoft Corporation	
CLBCatQ.DLL	COM+ Configuration Catalog	Microsoft Corporation	
comctl32.dll	Bibliothek für Steuerelemente	Microsoft Corporation	
common_odb			
comphelp4MSC.dll		OpenOffice.org	
COMPSTUI.dll	DLL für die Benutzeroberfläche der...	Microsoft Corporation	
configmgr2.uno.dll		OpenOffice.org	
cppcanvasmi.dll		OpenOffice.org	
cpu3.dll			
cpuhelper3MSC.dll		OpenOffice.org	
CRYPTBASE.dll	Base cryptographic API DLL	Microsoft Corporation	
CRYPTSP.dll	Cryptographic Service Provider API	Microsoft Corporation	
cuimi.dll		OpenOffice.org	
dbtoolsmi.dll		OpenOffice.org	

Image	Performance	Performance Graph	
Count: 9			
TID	CPU	Cycles Delta	Start Address
224		967.398	soffice.bin+0x13c5
228	28.531		sal3.dll!nt_cache_free+0x18a
232			gdipplus.dll!GdipCreateSolidFill+0x7e
236			MSVCR90.dll!lenthreadex+0x6f
3572			MSVCR90.dll!lenthreadex+0x6f
1540			ntdll.dll!RtlIsCriticalSectionLockedE
3256			MSVCR90.dll!lenthreadex+0x6f
2584			MSVCR90.dll!lenthreadex+0x6f
3680			ntdll.dll!RtlRegisterThreadWithCsrs

Process	PID	CPU	Description	Company Name
firefox.exe	988		Firefox	Mozilla Corporation

Name	Description	Company Name	Version
ACE.dll	Adobe Color Engine	Adobe Systems Incorporated	2.17.1.1
AzoreRd32.dll	Adobe Reader 9.3	Adobe Systems Incorporated	9.3.2.163
actxproxy.dll	ActiveX Interface Marshaling Library	Microsoft Corporation	6.1.7600.16385
adblockplus.jar			
ADVAPI32.dll	Erweiterte Windows 32 Base-API	Microsoft Corporation	6.1.7600.16385
AGM.dll	Adobe Graphics Manager	Adobe Systems Incorporated	4.18.88.1
Annots.api	Adobe Acrobat Annot Plug-in	Adobe Systems Incorporated	9.3.2.163
Annots.DEU	Adobe Acrobat Annot Plug-in	Adobe Systems Incorporated	9.1.0.163

Software ist ein Sammelbegriff für die Gesamtheit ausführbarer Programme und zugehörige Daten, wird aber auch für bestimmte Programmpakete verwendet, z.B. ERP-Software.

Umgangssprachlich werden ganze Programmpakete wie Microsoft Word als Programm bezeichnet. Im Vortrag meint Programm eine einzelne ausführbare Datei, z.B. C:\Program Files\Mozilla Firefox\firefox.exe oder /bin/sh. Der Vortrag unterscheidet nicht zwischen Hauptprogramm und Bibliotheken.

In der Informatik ist ein Prozess ein im Hauptspeicher ausgeführtes Programm. Für den Vortrag sind Ausführungszustände und Threads nicht relevant.

Fehler, Angreifbarkeit, Schwachstelle

- **Jede Software hat Fehler**
 - statistisch 2 Fehler pro 1000 Zeilen Programmcode
- **30% der Fehler eröffnen Angriffsmöglichkeiten**
 - neue Fehlerklassen werden entdeckt
- **Bekannte Schwachstellen werden ausgenutzt**
 - Exploit-Code spätestens nach 30 Tagen verfügbar

Lesenswerte Querverweise:

- Fehler-TOP25: <http://www.sans.org/top25-programming-errors/>
- MITRE CVE: <http://cve.mitre.org/>
- First CVSS: <http://www.first.org/cvss/>
- Fehlerklassen: <http://de.wikipedia.org/wiki/Kategorie:Sicherheitslücke>
- Webanwendungen: <http://de.wikipedia.org/wiki/Webanwendung#Sicherheit>
- Programmfehler: <http://de.wikipedia.org/wiki/Programmfehler>
- Pufferüberlauf Heap: <http://heise.de/-270800>
- Schwachstellen: <http://www.viruslist.com/de/hackers/info>

Proof-of-Concept, Exploit, Angriffe

■ Zeitfenster wird immer kürzer

- gute Hacker brauchen nur wenige Tage
- Exploit-Code kann automatisiert erzeugt werden
- Sicherheitshinweis nachdem ausgenutzt wird

■ Ausnutzung ist ein Geschäft

- Bot-Netz-Frameworks ab 200 US\$
- Werbung, Erpressung, Raubkopie-Vertrieb
- Industrie- und Wirtschaftsspionage?

Qualys hat anonymisierte Daten ihrer Kunden analysiert und die Ergebnisse auf <http://www.qualys.com/research/rnd/vulnlaws/> und der RSA-Konferenz vorgestellt: https://www.eventbuilder.com/main/event_desc.asp?eventid=k6b6g8p0

- 30 Tage nach Bekanntwerden kritischer Schwachstelle sind im Schnitt erst 50% der betroffenen Systeme gepatcht. Hierbei gibt es erhebliche Unterschiede je nach Branche und aufgrund der Eigenheiten der dort eingesetzten Systeme.
- Jährlich werden 12 der 20 kritischsten Schwachstellen durch noch drängendere ersetzt.
- 90% aller Schwachstellen werden nie vollständig behoben. Einige bleiben immer in der TOP20, z.B. MS Office XP/2003/2007 XML, Windows 2003 SP2, Adobe Acrobat, Oracle/Sun Java Plug-in.
- 30% der insgesamt gefundenen Schwachstellen können dazu genutzt werden, Kontrolle über den verwundbaren Computer zu erlangen.
- Für 80% aller Schwachstellen wird binnen 60 Tagen ein Exploit geschrieben und veröffentlicht. Microsoft geht davon aus, dass spätestens 30 Tage nach der Veröffentlichung eines Patches auf jeden Fall ein Exploit zur Verfügung steht.
- Bei privaten/HomeOffice-Installationen wird Windows-Update im Schnitt erst nach einem Jahr (erneut) ausgeführt, Office-Update erst nach 5 Jahren.
- 2009 wurden mehr als 40% der durch Webanwendungen verursachten Einbrüche in PCs über Exploits für Adobe Reader ausgeführt.

Phishing, Drive-by-Download, Botnet

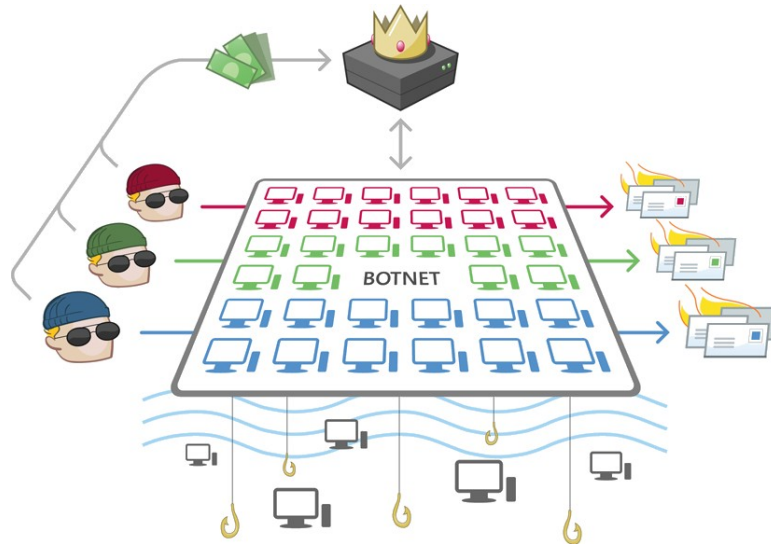


Bild: Waledac-Takedown, Quelle: [Microsoft Technet Blog](http://microsoft.com/technet) vom 25.02.2010
 Heise-Artikel hierzu: <http://heise.de/-940227>, <http://heise.de/-943084>

Waledac nutzt bisher klassische Phishing-Methoden wie Email-Anhänge und verleitet den Benutzer, diese auszuführen. Dies hat zunächst nichts mit Updates zu tun. Allerdings nutzen neuere Botnetze verstärkt im Vorfeld gekaperte Server oder in der Email eingebettete Flash/PDF-Dateien, um auch ohne „Anklicken des Anhangs“ zum Ziel zukommen.

- Internet Explorer 0-day: <http://heise.de/-189633.html>
- Phishing: <http://de.wikipedia.org/wiki/Phishing>
- Drive-by-Download: <http://de.wikipedia.org/wiki/Drive-by-Download>
- Botnet: <http://de.wikipedia.org/wiki/Botnetz>
- BBC Botnet Takeover: <http://heise.de/-206311>

Sicherheitsaktualisierungen

	2008	2009	2010
Sun Java Runtime	3	3	1
Adobe Reader	5	9	2
Adobe Flash Player	6	5	2
Apple Quicktime	5	3	1
Apache httpd	3	3	1
OpenSSL	1	2	1
ISC BIND	2	3	1
Blackberry Enterprise Srv	8	1	7
Microsoft Windows	14	13	6
debian Linux (alle Pakete!)	251	269	67

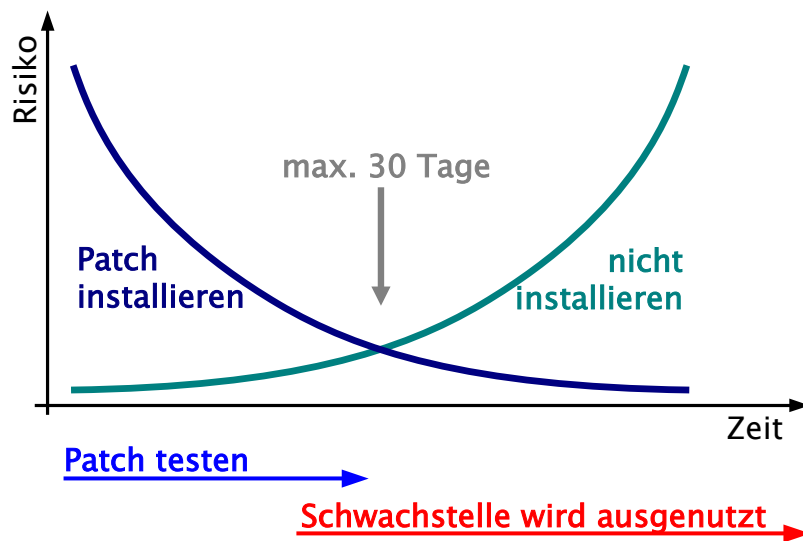
Quelle: Webseiten / Security Advisories der Hersteller

Die Zahlen bei den Client- und Server-Applikationen beziehen sich auf die jeweils aktuelle Version der Anwendung, wobei gleichzeitig erschienene Advisories für mehrere Versionen nicht doppelt gezählt wurden.

Die Zahlen der letzten beiden Zeilen (Betriebssysteme) sind erklärungsbedürftig:

- bei Microsoft wurden nur die regulären und außer-der-Reihe Patchtermine gezählt, nicht die Zahl der der Advisories oder die behobenen Schwachstellen.
- bei Linux wurden alle Pakete der Distribution gezählt unabhängig davon, ob
 - sie zum Betriebssystem oder zu einer Applikation gezählt werden müssten
 - sie typischerweise installiert sind (oder parallel installierbar sind)
 - mehrere Patches an einem Tag erschienen sind

Aktuell sicher oder stabil unsicher?



- Aktualisierungen für typische Programme erscheinen unregelmäßig aber relativ oft (im Schnitt jede Woche).
- Angriffe sind heute fast immer mehrstufig: Social Engineering, Phishing, Angreifbarkeit von Software, schwache Passwörter.
- 2009 gab es mehrere prominente Angriffe aufgrund verschiedener Fehler, heute oft nicht aktuelle Software + Phishing
- Hersteller benötigen für die Qualitätssicherung Zeit, um alle betroffenen Versionen und Wechselwirkungen mit anderen Programmen zu testen.
- Daher erscheinen Patches oft erst NACHDEM die Schwachstelle bereits in größerem Umfang aktiv ausgenutzt wird.
- Dem Administrator bleibt wenig Zeit, die verfügbaren Aktualisierungen in SEINER Umgebung zu testen und einzuspielen.
- Eine Risikoanalyse sollte darüber entscheiden, ob Teilfunktionalitäten für die Dauer der Angreifbarkeit abgeschaltet/blockiert werden.

Rahmenbedingungen

■ Vernetzung

- keine Trennung von Büro, Produktion und WLAN
- Firewall verhindert keine Angriffe „von innen heraus“

■ Vermischung geschäftlich und privat

- Arbeiten im HomeOffice
- Smartphone: Kalender, Kontakte
- Leben im Web 2.0 / Soziale Netzwerke

Die Vernetzung schreitet immer weiter fort, nicht nur zwischen Filialen und Anwendungen innerhalb der Firma („Integration“), sondern auch innerhalb des Konzerns (z.B. Abtrennung von Geschäftsbereichen), zwischen Partnern (z.B. Vertriebsbeauftragte) oder in Form von Outsourcing (SaaS, cloud based computing).

Die eigentliche Herausforderung stellen jedoch die Anbindung mobiler Geräte (z.B. Scanner im Lager) per WLAN, von Smartphones wie Blackberry oder iPhone an die Kommunikations- und CRM-/ERP-Systeme sowie die zunehmende Vermischung von privaten und geschäftlichen Daten (z.B. Kontakte) dar.

Es fällt schwer, eine klare „Firmengrenze“ zu definieren: Angriffe erfolgen heute typischerweise nicht durch die Vordertür (Firewall), sondern durch die Hintertür (Email, IM, MMS, ...).

Wenn man am Konzept Firewall festhält, dann ist es inzwischen nötig, diese so zu erweitern, dass Sie auch Prozesse auf den Clients und ausgehenden Verkehr begutachtet.

Visionäre sprechen jedoch bereits davon, die Firewall abzuschaffen, da sie zu unflexibel und zu teuer sei. Wenn man dies tut, so muss klar sein, wie der bisher dadurch gegebene Schutz in Zukunft gewährleistet wird.

Ein Grundbaustein davon ist die Robustheit des einzelnen Systems bzw. der einzelnen Anwendung, d.h. Verzicht (im Sinne einer Reduktion der Softwarekomplexität), Hardening (kleinere Angriffsfläche durch minimal laufende Anwendungen) und Patchmanagement sind dann unbedingt erforderlich.

Handlungsmöglichkeiten

■ Prinzipielle Alternativen

- Isolation
- Technische Schutzvorkehrungen
- Übergangslösung
- Software aktualisieren
- Versionswechsel
- alternatives System verwenden

■ Information und ggfs. Schulung der Anwender

■ Nachvollziehbarkeit der Änderungen

- Isolation = „Stand-Alone-PC/Server“ (ohne Netzwerkverbindung)
- Schutzmechanismen siehe nächste Folie
- Übergangslösung/Workaround
 - Abschalten bestimmter Funktionen
 - mit geringen Auswirkungen: z.B. Setzen von KillBits
 - mit massiven Auswirkungen: z.B. Filtern aller EXE-/PDF-Dokumente
- Software aktualisieren
 - heute meist Ersetzen einzelner Dateien, aber auch Modifikation möglich
 - Software Lifecycle-Management beginnt aber schon früher:
 - Vereinheitlichung durch automatisierte Provisionierung neuer PCs
 - Self-Service aus Software-Bibliothek (Baukastensystem/Lizenzmanagement/Freigabeverfahren)
 - Software neu installieren (uninstall, purge, install)
- alternative Systeme verwenden
 - z.B: FreeBSD, Solaris
 - z.T. leichter angreifbar (Mac OS X), aber nicht im Ziel der Massenhacks

Technische Schutzmechanismen

■ Netzwerk-Filter

- Firewall, Intrusion Prevention System, Reverse-Proxy

■ Server-Schutzmechanismen

- Hardening/Stripping, Wrapper, Privilege Separation

■ Absicherung der Clients

- „Virenschanner“ (heute FW+HIDS+AV+AR+AP+AS+...)
- NoScript/RequestPolicy
- keine Adminrechte

Greifen die ergriffenen Maßnahmen auch außerhalb der Firma?

- Smartphone
- HomeOffice-Arbeitsplatz
- Laptop-Verschlüsselung
- Umgang mit USB-Sticks, CDs/DVDs, Speicherkarten
- Handy-Kamera und -speicher
- Email-Verschlüsselung
- Äußerungen in sozialen Netzwerken

Einbettung

■ Software Lifecycle Management

- Softwarearchiv
- Einführung
- Schulung
- Patchmanagement
- Ausphasen

■ Security-Management

■ Changemanagement

Patchmanagement kann separat betrieben werden. In der Regel gibt es aber schon Prozesse und Systeme, die diese unterstützen. Um den (zusätzlichen) Aufwand so gering wie möglich zu halten (und sicherzustellen, dass nichts vergessen wird), ist es sinnvoll, auf vorhandene Abläufe aufzusetzen.

Beispiele:

- Ein Admin, der alle Aufträge und ToDos in seiner Email-Inbox verwaltet, sollte sich Advisories und Statusänderungen der Server per Email schicken lassen.
- Wenn bereits ein Trouble-Ticket-System vorhanden ist, sollten Advisories und Update-Verfügbarkeitsmeldungen der Server automatisch als Tickets erzeugt und „normal“ abgearbeitet werden. Sicherlich ist es dabei sinnvoll, Prioritäten und Eskalationsstufen anzupassen.

Security-Management oder Change-Management sind als Prozess (z.B. im Rahmen von ITIL oder ISO 27000) eher in großen Firmen zu finden.

Aber auch in kleinen Firmen muss klar sein, wer für die Sicherheit (hier: Security Patches) verantwortlich ist.

Kennntnisnahme / Prozess–Start

■ Sicherheitshinweise

- heise Security, Bürger–CERT, DFN–CERT
- Secunia, Qualys, SecurityFocus BugTraq

■ Scannen

- Heise Update, Secunia OSI, IT.shavlik.com
- MBSA, WSUS, Shavlik NetChk Protect, Secunia CSI

■ Autoupdate

- Benachrichtigung des Serveradmins
- Hinweis an den PC–Benutzer

Es gibt viele Informationsquellen. Zu viele. So viele, dass man sie unmöglich alle lesen kann. Daher ist es wichtig, sich eine Informationsquelle zu suchen, die dem eigenen Niveau („Wie technisch detailliert soll es sein?“) entspricht und eine Bewertung (wegen konkurrierender Ziele) für die eigenen Systeme liefert.

Um einen Status Quo ermitteln zu können ist ein initiales Scannen unerlässlich. Hierbei werden auch längst vergessene Anwendungen und Sicherheitslücken gefunden. Eingebauten Selbstaktualisierungsmechanismen finden diese nicht.

Informationsquellen

- heise security <http://www.heise.de/security/>
- Bürger–CERT <http://www.buerger-cert.de/>
- Mailingliste <http://www.patchmanagement.org/>

Scanner

- Secunia CSI http://secunia.com/vulnerability_scanning/corporate/
- IT.shavlik.com <https://it.shavlik.com/>
- Microsoft Tools <http://technet.microsoft.com/security/>
(Baseline Security Analyzer, Security Assessment Tool)

Für private Nutzung kostenlose Scanner

- Secunia PSI http://secunia.com/vulnerability_scanning/personal/
- UpdateChecker <http://www.filehippo.com/updatechecker>
(nicht nur sicherheitskritische Updates)

Technische Warnung des Bürger-CERT

[Bcert-2010-0033/1]: Adobe Reader und Acrobat 9.3.2 und 8.2.2

Risiko: Sehr hoch

Betroffene Systeme:

Adobe Reader und Acrobat vor Version 9.3.2 (Windows, Unix, Mac)
Adobe Reader und Acrobat vor Version 8.2.2 (Windows, Unix, Mac)

Empfehlung:

Das Bürger-CERT empfiehlt die zeitnahe Installation der von Adobe bereitgestellten Sicherheitsupdates. Dies geschieht am einfachsten über den Menüpunkt "Hilfe" und die Option "Nach Updates suchen...".

Beschreibung:

Ein entfernter Angreifer kann mehrere Schwachstellen in Adobe Reader und Acrobat ausnutzen, um einen Absturz der Anwendung auszulösen oder um Schadsoftware auf dem System des Opfers auszuführen.

Dazu ist nicht zwingend das Öffnen einer PDF-Datei notwendig, es kann der Besuch einer manipulierten Webseite bei installiertem Adobe Reader / Acrobat Plug-in genügen.

(Gekürztes) Beispiel einer Email des Bürger-CERT zur aktuellen Schwachstelle des Adobe Readers. Deren Emails versuchen in der Regel, keine technische Erklärung sondern beschreiben konkret in möglichst einfachen Worten, was ein Endnutzer tun muss.

Zur Beurteilung sind diese Emails nicht geeignet, nur als Hinweis, dass etwas getan werden muss. Auch ist zu berücksichtigen, dass typische Unternehmensanwendungen wie SAP oder Exchange aufgrund der Zielgruppe hier nicht thematisiert werden.

Technisch detaillierter und zur Beurteilung in der Regel hilfreich sind die Emails des DFN-CERT, welche aber primär an deren Mitglieder verschickt werden. Vor Kurzem hat das DFN-CERT aber ein öffentlich erreichbares Archiv angelegt: <https://portal.cert.dfn.de/adv/archive/>

Daneben bieten kommerzielle Anbieter wie SecurityFocus, Qualys, Secunia, und viele andere Dienste an, die nur Warnungen für die beim Kunden eingesetzten Systeme und eine Bewertung liefern.

medien forum freiburg

PR IT SI

Inventar-Scanner

IT: shavlik.com

Log Out Account Store

nursoda

Network (1)

Machines (3)

Virtual (0)

Software (93)

<Home : Machines

Export: XLS CSV

Drag IT - drag a column here to group by that column

fix IT	Machine Name	VM	OS	Status	Last Patch S	Last Asset S	Last Deploy
<input type="checkbox"/>	ERDE				11.04.2010 17:12	11.04.2010 17:12	
<input type="checkbox"/>	NEGOTIATIS				14.04.2010 15:57	14.04.2010 15:57	11.04.2010 15:53
<input type="checkbox"/>	ZAPHOD				11.04.2010 18:07	11.04.2010 18:07	11.04.2010 17:58

find IT.
click to start

fix IT.

forget IT.

Reports

[Scan History](#)

[Missing Patches \(0\)](#)

[Patch Deployment History](#)

Software aktuell halten - 15.04.2010 - © Olav Seyfarth

14

Der cloud-basierte Scanner von Shavlik benötigt eine .NET 3.5-Umgebung und lädt einen entsprechenden Agenten auf das Zielsystem. Sofern über IPC\$ Zugriff auf andere Computer im Netz besteht, können auch diese gescannt werden.

Die ersten 10 IPs und 100 Scans sind hierbei kostenlos möglich, darüber hinaus gehende Leistungen können flexibel gebucht werden.

Besonders an dieser Lösung ist die Möglichkeit, nicht nur zu scannen, sondern gefundene Schwachstellen mit „einem Klick“ auf allen gescannten Maschinen beheben zu lassen. Dies ist in der Regel nur für kleinere Netzwerke oder bei regelmäßiger Anwendung wünschenswert.

Für den Fall, dass ein PC durchgeführten Updates neu gestartet werden muss, bekommt der Endanwender einen Hinweis, mit dem er den Zeitpunkt des Neustarts selbst festlegen kann.

Tägliches Update eines Linux-Servers

```
(cron-daily) # apt-get update ; apt-get dist-upgrade

Get:1 http://ftp.debian.org testing Release.gpg
Get:2 http://ftp.debian.org testing/main Packages/DiffIndex
Get:3 http://ftp.debian.org testing/main 2010-04-14-1509.58.pdiff

Reading package lists
Building dependency tree
Reading state information

The following packages will be upgraded: iputils-ping libgd2-xpm

Get:1 iputils-ping 3:20100214-1 [53.0kB]
Get:2 libgd2-xpm 2.0.36~rc1~dfsg-3.2 [229kB]
(Reading database ... 34528 files and directories currently installed.)
Preparing to replace iputils-ping 3:20071127-2
Unpacking replacement iputils-ping
Preparing to replace libgd2-xpm 2.0.36~rc1~dfsg-3
Unpacking replacement libgd2-xpm
Processing triggers for man-db
Setting up iputils-ping (3:20100214-1)
Setting up libgd2-xpm (2.0.36~rc1~dfsg-3.2)

2 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Bei UNIX-/Linux-basierten Server-Systemen können Aktualisierungen weitgehend automatisiert werden. In dieser (gekürzten) cron-Email eines debian-Systems wurden zwei Pakete automatisch durch neuere Versionen ersetzt.

Falls Updates automatisch eingespielt werden, sollte der Administrator nach der Aktualisierung unbedingt das Protokoll prüfen.

Spielarten auf Client-Systemen

■ Person

- Benutzer aktualisiert selbst
- Admin aktualisiert mit (Remote-)Login
- (Zentrale) Software aktualisiert im Hintergrund

■ Hilfsmittel

- AutoUpdate Betriebssystem / Programm / AddOn
- Spezieller Updater des Herstellers/Installers
- Distribution aktualisiert enthaltene Programme
- Spezielles System zur Softwareverteilung

Nicht-repräsentative Auswahl einiger Hersteller von Patchmanagement-Lösungen im Bereich Client und Server, z.T. nur für Windows:

- Shavlik: <http://www.shavlik.com/> (online, standalone und SCCM)
- Secunia: <http://www.secunia.com/> (Agent, Aktualisierung via WSUS)
- Empirum: <http://www.matrix42.de/>
- LANdesk: <http://www.landesk.de/>
- Altiris: <http://www.symantec.com/business/client-management-suite>
- OCS Inventory: <http://www.ocsinventory-ng.org>

Herausforderungen Server

■ Herausforderungen

- Verfügbarkeit, Downtime minimieren
- verschobener Updates nicht vergessen
- Konfiguration sichern / Änderungen dokumentieren
- Scannen aller Systeme sinnvoll

■ virtuelle Server

- Virtualisierungs-Software selbst nicht vergessen
- VMware, Xen, ...: mehrere kritische Fehler pro Jahr

Server-Systeme haben in der Regel hohe Anforderungen an die Verfügbarkeit, da sie von außen erreichbar sind, Basisdienste für andere Services im Unternehmen zur Verfügung stellen oder die Produktion direkt von ihrer Stabilität abhängt.

Mehrere Gründe sprechen dagegen, Systeme zu patchen

- „Never touch a running system“: Veränderung am System führen in der Regel zu verändertem Verhalten des Systems und sind per se nicht förderlich für die Stabilität.
- Die Qualität „schneller“ Sicherheits-Updates entspricht oft nicht der sonst gewohnten Qualität der Software.
- Um Auswirkungen von Updates auf die eigene Umgebung aussagekräftig testen zu können, ist eine eigene Testumgebung nötig, die der echten Anwendung hinsichtlich Komponenten und Daten möglichst nahe kommt.
- Das System ist beim Update in der Regel für eine gewisse Zeit nicht erreichbar. Lösungen, die dies umgehen, sind aufwändig und teuer.

Daher liegt es nahe, getestete Aktualisierungen erst in einem regelmäßig geplanten „Patchfenster“ durchzuführen. Aber auch dagegen gibt es Argumente:

- Es besteht die Gefahr, dass die Schwachstelle bis dahin bereits ausgenutzt wird und das System (auch von intern!) betroffen ist.
- Verschobene Updates gehen nicht selten im Administrationsalltag unter.
- Mehrere Updates auf ein Mal durchzuführen erschwert die Eingrenzung von Fehlern und die Zeit bis zur Wiederherstellung der Verfügbarkeit des Systems.

Sicherheits- & Netzwerkkomponenten

■ weitreichende Angreifbarkeit

- schnelle Updates nach Release nötig
- Abhängigkeiten anderer Systeme
- Abhängigkeiten von anderen Systemen

■ Dokumentation

- automatischer Vergleich der Konfiguration zur Erkennung von Manipulationen
- Rollback der Änderungen bei Fehlern

Netzwerkkomponenten haben sehr unterschiedliche Priorität und können nicht über einen Kamm geschoren werden:

- Sicherheitskomponenten wie Firewalls, Proxies und Loadbalancer sind kritisch, weil sie zentrale Zugangs- und somit Fehlerquellen darstellen. Meist sind alle dahinter liegende Komponenten unmittelbar in ihrer Funktion eingeschränkt.
- Für Netzwerkinfrastruktur wie Router, Switches und AccessPoints gilt ähnliches, da z.B. die Annahme von Waren ohne Scanner erheblich länger dauert, wenn die Handscanner nicht benutzt werden können. Sofern die Produktion über JIT oder Kanban-Prozesse von den üblichen Durchlaufzeiten abhängt, haben Störungen direkte Auswirkungen auf die Produktivität.
- Aber selbst netzwerkfähige Komponenten wie Drucker/Fax/Kopierer müssen berücksichtigt werden. Zwar kann in der Regel auf alternative Ressourcen zugegriffen werden, es gibt aber auch Schadsoftware, Fehler in deren IP-Stacks oder embedded Webserver ausnutzt. So kann ein Drucker ein LAN lahmlegen.

Durch neue Firmware (analog zu neuen Versionen im Bereich Server-Software) werden z.T. auch veränderte Konfigurationen benötigt. Die Kontrolle über die Integrität der Konfiguration hat in diesem Zusammenhang eine große Bedeutung.

Differenzmeldungen aus automatischen Vergleichen können im Changemanagement anhand der geplanten und freigegebenen Änderungen gefiltert werden.

Dies ist fürs Patchmanagement relevant, weil mit dieser Methode nicht nur die Konfiguration, sondern auch die Softwarestände bestimmt werden können.

Embedded Systeme berücksichtigen

- **Netzwerkanbindung = Firmwareupdates!**
 - Herstellergarantie auch für Sicherheitsupdates?
 - Isolation, wenn kein Firmwareupdate möglich ist
- **Komponenten von Drittanbietern**
 - zugekauft (z.B. IP-Stack) oder OpenSource (z.B. SSL)
 - Updates für im Produkt verwendete Komponenten?
- **Drucker, Scanner, Kopierer, Faxgeräte**
 - LAN-Stack eines HP-Druckers legte LAN lahm

Hersteller intelligenter Sensoren/Aktoren setzen Standardkomponenten ein. Spätestens mit Industrial Ethernet besteht die Gefahr, dass für die Produktionsprozesse oder gar für die Sicherheit von Leib und Leben wichtige Systeme auch eine Verbindung ins Firmen-LAN besteht. Hier hilft nur eine saubere Trennung oder Abschottung.

Sofern diese Systeme jedoch im LAN betrieben werden sollen (Beispiel: IP-Video-Kamera), müssen diese Systeme ebenfalls im Rahmen des Patchmanagements betrachtet werden. Denn auch eine Videokamera hat einen Webserver eingebaut, nicht selten mit Unterstützung für Skriptsprachen. Eine solche Kamera ist wie ein vollwertiger Webserver zu betrachten und abzusichern.

Dies setzt voraus, dass die Software eines Systems aktualisiert werden kann (beschreibbarer nichtflüchtiger Speicher für die Programmierung des Geräts) und aktualisiert werden darf (z.B. falls eine Sicherheitszertifizierung nötig ist).

Schwierig wird es auch, wenn der Hersteller selbst Softwarebausteine hinzukaufft oder Open-Source-Implementierungen verwendet, diese aber nicht oder nicht zeitnah aktualisiert.

Sichere Softwareentwicklung



4 Sicheres Design

4.1 Grundlegende Anforderungen an das Design

Die folgenden Grundregeln sollten Sie unabhängig von Implementierungstechnologie und Anwendungsszenario befolgen.

Sorgen Sie für Sicherheit im Versagensfall
Wenn Ihre Anwendung abstürzt, dürfen keine vertraulichen Daten preisgegeben werden. Verfassen Sie eine aussagekräftige, aber allgemein formulierte Fehlermeldung. Sie darf keine internen Systeminformationen enthalten, die einem Angreifer eventuelle Schwachstellen in Ihrer Anwendung aufzeigen würden.

Vertrauen Sie nicht auf „Security by Obscurity“
Der Versuch, durch Vertuschen von Schwachstellen die Sicherheit Ihrer Anwendung zu verbessern, kann langfristig nicht funktionieren – und auch die kurzfristige Wirkung ist keinesfalls garantiert. In der Vergangenheit wurden viele, solcher vermeintlich geschützten Systeme erfolgreich angegriffen.

Handeln Sie nach dem Prinzip der minimalen Rechtevergabe
Verfahren Sie nach dem Prinzip der minimalen Rechtevergabe: Verwenden Sie Benutzerkonten mit den geringstmöglichen Privilegien und Zugriffsrechten.

Bauen Sie mehrere Verteidigungsmechanismen ein
Verwenden Sie mehr als einen Sicherheitsmechanismus. Bedenken Sie, dass eines Ihrer Verfahren geknackt werden könnte.

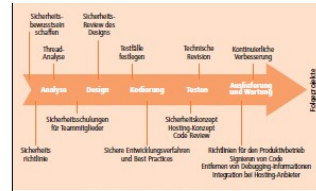


Abbildung 1: Sicherheit bei der Softwareentwicklung

- **Analyse**
 - Definieren Sie die Sicherheitsziele Ihres Produkts
- **Design**
 - Sichere Anwendungsarchitektur
 - Design Review
- **Kodierung**
 - Sicheres Kodieren
 - Code Review
- **Testphase**
 - Angriffssimulationen
- **Auslieferung und Wartung**
 - Sichere Standardkonfiguration
 - Wartungsplan
 - Zeiträume für das Patch-Management
 - Kriterien für die Fehlerbehebung (Bug-Fixing), besonders bei sicherheitsrelevanten Fehlern

Wäre Software sicher, bräuchte man kein Patchmanagement. Ist sie aber nicht. Sofern Sie selbst Software entwickeln (lassen), sollten Sie zunächst alles unternehmen, um Fehler im Vorfeld zu vermeiden, aber eben auch daran denken, was passiert, wenn doch ein (sicherheitskritischer) Fehler in Ihrer Software (oder einer verwendeten Komponente eines Drittanbieters) auftaucht.

Einen immer noch aktuellen Überblick gibt die Broschüre des DsiN e.V. von 2005: Pocket-Seminar: <https://www.sicher-im-netz.de/unternehmen/163.aspx>

Microsofts Best-Practice-Ansatz „Security Development Lifecycle“ (SDL) ist ein weltweit anerkannter de-facto Standard für sichere Softwareentwicklung. Am 01.04.2010 erschien die Version 5.0. Es gibt auch eine

Microsoft SDL: <http://www.microsoft.com/security/sdl/>
<http://msdn.microsoft.com/de-de/cc766739.aspx>
<http://msdn.microsoft.com/de-de/cc188938.aspx>
<http://heise.de/-403663>

Die Top 25 der Programmierfehler, veröffentlicht von MITRE und SANS: MITRE Top 25: <http://cwe.mitre.org/top25/>

Die Top 10 der Sicherheitslücken für Webentwickler, veröffentlicht vom Open Web Application Security Project

OWASP Top 10 <http://www.owasp.org/>

Security Engineering

■ Sicherheit als (begleitender) Prozess



■ Bestandteil der funktionalen Spezifikation

- unabhängig von der Entwicklungsmethodik
- Entwicklungstools nutzen, Bibliotheken, Fuzzing

■ Sicherheitsverantwortliche benennen

Wie so oft ist auch Security Engineering Chefsache. Sie muss nicht nur vom Programmierer umgesetzt werden, sondern der gesamte Entwicklungsprozess muss Sicherheit als roten Faden aufgreifen.

Entwickler kennen sich mit den Methoden des Software-Engineering aus. Requirements-, Quality-, und Security-Engineering stehen jedoch noch immer nur selten auf dem Lehrplan. Diese Disziplinen sind nötig, um nicht nur eine funktional korrekte, sondern auch eine dauerhaft robuste Software entwickeln und pflegen zu können. Eine klare Verantwortlichkeit (mit Kompetenzen!) und die gezielte Schulung dieser Verantwortlichen ist nötig!

Das Thema sichere Softwareentwicklung füllt selbst mehrere Vorträge und ist nicht Kernkompetenz des Autors. Daher sei hier auf einige zufällig ausgewählte Vorlesungen und Schulungsanbieter in diesem Bereich verwiesen:

UNI Freiburg: <http://www.telematik.uni-freiburg.de/>

UNI München: <http://www.sec.in.tum.de/security-engineering-ss10/>

Secorvo College: <http://www.secorvo.de/college/seminare/>

Cirosec Training: <http://www.cirosec.de/deutsch/trainings/>

Virtual Forge: http://www.virtualforge.de/isseco_training.php

Recurity Labs: <http://www.recurity-labs.com/content/trainings/>

Softwarewartung

■ Sicht des Herstellers

- Fehlerbehebung für aktuelle und ältere Versionen
- Wartungszeitraum festlegen und mitteilen
- auch für gekaufte und Open-Source Komponenten

■ Sicht des Kunden

- Lieferant zur Fehlerbehebung verpflichten
- keine Software ohne Wartungsvertrag einsetzen
- Support auch bei Abhängigkeit von Drittsystemen?

Klar abgegrenzte Quellcode-Bäume bilden die Basis von Software-Releases. Mit jedem Release wächst aber der Aufwand bei der Behebung (sicherheitskritischer) Fehler (und beim Support). Es ist daher verständlich, dass man als Hersteller auf Dauer nicht jede jemals gelieferte Version warten kann.

Daher ist es wichtig, bereits bei der Produktentwicklung festzulegen, wie lange ein Produkt gepflegt wird. Es empfiehlt sich, hierbei zwei Zeiträume festzulegen:

Während mit der Veröffentlichung der Folgeversion oft kaum noch funktionale Erweiterungen in der nun „alten“ Version implementiert werden, erwartet der Kunde, dass er Fehlerbehebungen noch für einen längeren Zeitraum erhält.

Oft vergessen wird dabei, dass zugekaufte Komponenten wie Netzwerkstack (IP) oder Protokollimplementierungen (SSL) selbst Fehler enthalten und damit direkt das eigene Produkt anfällig machen. Der Zulieferer muss zeitnah auf neu erkannte Schwachstellen reagieren und alle Produkte mit der verwundbaren Komponente müssen aktualisiert werden (können).

Für den Einkauf beim Kunden bedeutet dies, dass jede Software nur so lange eingesetzt werden darf, wie diese vom Hersteller Sicherheits-Updates erhält.

Zu jeder Software sollte ein Wartungsvertrag existieren, der den Hersteller verpflichtet, Sicherheitslücken zeitnah zu stopfen.

Um den Aufwand gering zu halten und sicherzustellen, dass Sicherheitspatches tatsächlich installiert werden, ist es sinnvoll, die bereits vorhandenen Prozesse zu nutzen, z.B. als Incident im Change Management (ITIL) oder im TT-System.

Kommunikation bei Schwachstellen

■ Hinweise von Dritten

- Kontaktdaten veröffentlichen, z.B. security@...
- Ablauf beschreiben („responsible disclosure“)

■ Information an Kunden

- Vorlage für Sicherheitsmitteilung erstellen

■ Umgang mit Presse

- Mitarbeiter einweisen: wer darf was beantworten

Unter Hackern gibt es verschiedene Überzeugungen was die Veröffentlichung von neu gefundenen Schwachstellen angeht:

- verantwortungsvolle Veröffentlichung (responsible disclosure)
 - Kontaktaufnahme mit dem Hersteller
 - Veröffentlichung gemeinsam mit Patch des Herstellers (credits)
- unbedingte Veröffentlichung (full disclosure)
 - Information des Herstellers mit Hinweis, wann Veröffentlichung erfolgt oder auch gar keine Benachrichtigung des Herstellers
 - Veröffentlichung auf den „üblichen Listen“ oder dem eigenen Blog
- keine Veröffentlichung (Oday underground market)
 - keine Benachrichtigung des Herstellers
 - Keine Veröffentlichung

Aufgrund der immer kürzer werdenden Zeit zwischen dem Bekanntwerden einer Schwachstelle und der Verfügbarkeit eines modular nutzbaren Exploits sollte ein Hersteller es einem wohlgesonnenen Hacker leicht machen, einen kompetenten Ansprechpartner (den benannten Sicherheitsverantwortlichen) zu kontaktieren. Die Vergangenheit hat gezeigt, dass Hacker durchaus versucht haben, Kontakt aufzunehmen. Wenn diese jedoch keine Reaktion erhielten veröffentlichten sie kurzerhand ohne dass der Hersteller vorbereitet war.

Ebenso wichtig ist es, Krisenkommunikation zu planen und zu üben.

Patchmanagement scheitert an

- **Softwarevielfalt**
 - Mehrere Programme für eine Aufgabe
 - Verschiedene Versionen desselben Programms
 - Benutzer brauche/wollen Administratorrechte
- **Abhängigkeit von bestimmten Versionen**
 - sonst keine Funktion oder kein Support
- **Aufwand / Unkenntnis**
 - Administrator: Zeit
 - Geschäftsleitung: Geld

Sofern noch keine Prozesse zum Umgang mit Software im Unternehmen etabliert sind, scheitert die Einführung eines Patchmanagements häufig daran, dass es einen Software-Zoo (verschiedene Programme für ähnliche Aufgaben je nach Abteilung, etliche Versionen desselben Programms durch verschiedene Installationszeitpunkte und manuelle oder automatische Updates) gibt.

Darüber hinaus sind oft aus verschiedenen Zängen heraus Administrator-Rechte an Benutzer vergeben, sodass auch die Konfiguration der PCs selten einheitlich ist.

Hier ist zunächst eine Vereinheitlichung der Clients (z.B. durch Ablösung der Altsysteme) oder ein flexibles Werkzeug erforderlich, mit dem die verschiedenen Softwarekonfigurationen auf einen einheitlichen Stand gebracht werden können.

Technisch gibt es eine Reihe von Anwendungen, die eine bestimmte und daher nach kurzer Zeit veraltete Version einer Software von Drittanbietern voraussetzt. Ein typischer Kandidat hierfür ist Java, obwohl die Plattform ja eigentlich eine Unabhängigkeit garantieren sollte. Aber auch ein Windows Update kann SAP-Anwendungen aus dem Tritt bringen, wenn z.B. Internet Explorer aktualisiert wird.

Kritisch wird es dann, wenn ein Hersteller Support nur dann für sein Produkt gewährt, wenn z.B. eine ganz bestimmte Version des Betriebssystems installiert ist und so das Einspielen von (Sicherheits-)Aktualisierungen zum Verlust der Garantie bzw. des Supports führen würde. Hier muss der Kunde bereits beim Einkauf darauf achten, dass – nach zeitnaher Prüfung durch den Hersteller oder den Kunden selbst – zumindest sicherheitskritische Updates akzeptiert werden.

Probleme durch Aktualisierung

- **Softwarequalität leidet unter Zeitdruck**
 - Kompatibilitätsprobleme, Patch zum Patch
- **Verteilen von Schadsoftware in der Eile**
 - gefälschte Malware/Virens Scanner
- **Fehlende Transparenz bei Updates im Hintergrund**
 - PC langsamer, Benutzer weiß nicht warum
 - Update/Reboot zu für Benutzer ungünstiger Zeit
- **Server-Update mit Versionswechsel**
 - Migration der zuvor angepassten Konfiguration

Wären Autos von derselben Qualität, mit der heute Software ausgeliefert wird, dann gäbe es vermutlich jeden Tag einen Rückruf eines Herstellers. Software wird unter hohem Zeitdruck hergestellt. Bei Aktualisierungen kommt nun auch noch der Druck durch die Ausnutzbarkeit hinzu.

Aber auch der Administrator selbst ist unter Zeitdruck. In Gebieten, in denen er kein Experte ist, kann dies auch dazu führen, dass er einer gut gemachten Fälschung auf den Leim geht: die großen Sicherheitsdienstleister nennen bei den Gefahren durch Malware auch gefälschte Sicherheitssoftware als wesentliches Problem.

Leider wird der Endbenutzer bei der Frage nach Schutzprogrammen und Updates oft vergessen: er weiß meist nicht, was auf seinem Client oder dem Server, den er gerade nutzt, im Hintergrund passiert. Ausfall- und Wartezeiten, unerwartete Fehlermeldungen oder gar automatische Neustarts frustrieren die Anwender und tragen so dazu bei, dass diese Wege suchen (und finden), die angestoßenen Maßnahmen zu umgehen oder zu verhindern.

Gibt es für eine Software keine Aktualisierung mehr, kommt neben der Isolation oft ein Wechsel zur aktuellen Version in Betracht. Neue Versionen bringen jedoch meist auch eine neue Konfiguration mit sich. Der Zeitaufwand für die Migration der angepassten, im Laufe der Zeit optimierten Konfiguration (auf Clients und Servern) wird oft unterschätzt. Auch Testprozesse helfen hier nicht immer. Daher ist eine gute zeitliche Planung und die Einbindung oder zumindest Information aller Betroffenen nötig.

Empfehlungen

- **Vereinheitlichung der installierten Software**
 - Sonst höherer Aufwand für Updates (und Support)
- **Einteilung nach Systemtyp und Softwaretyp**
 - Netzwerkkomponenten, Firewalls, Virens Scanner
 - zum Internet offene Server
 - Internet-Applikationen inkl. Plug-Ins, evtl. Office
 - zentrale Systeme (Authentisierung, Datenbanken)
 - unternehmenskritische Software
 - selbstgeschriebene / wenig verbreitete Software

Es ist möglich, einen Standard für Clients, Server oder z.B. Handys zu definieren. Dieser Standard muss von Zeit zu Zeit überarbeitet werden. Möglicherweise ist es sinnvoll, Ableitungen vom Standard zu definieren, um den Anforderungen einzelner Nutzergruppen gerecht werden zu können.

Installations-Templates müssen selbst aktualisiert werden – sonst müssen alle Aktualisierungen direkt nach der Provisionierung und noch vor der Einbindung ins Netzwerk erfolgen.

Qualys schlägt vor, beim Patchmanagement verschiedene Cluster zu bilden, je nachdem, wie schnell aktualisiert werden muss und welche Auswirkungen der Ausfall eines Systems auf die Geschäftsprozesse hat. Die Zuordnung zu einer Gruppe ist nur anhand einer Risikoanalyse möglich, da es für die meisten Systeme Gründe gibt, dass Updates möglichst schnell gemacht werden müssen oder die Verfügbarkeit gewahrt bleiben muss.

Patchmanagement für Windows

- **AutoUpdate**
 - Microsoft, Adobe, Sun, Mozilla, ...
- **Online-Scanner/Updater**
 - IT.shavlik.com, Heise Security Update, Secunia OSI
- **WSUS / SCCM**
 - mit Feeds für Drittanbieter: Secunia / Shavlik
- **Inventory-/Patchmanagement-Software**
 - Shavlik NetChk Protect
 - Matrix42 Empirum
 - OCS Inventory NG

Für Windows ist die Angebotspalette an Patchmanagement-Software am größten, da die meisten Firmen Windows-basierte Arbeitsplatz-PCs managen müssen.

Gleichzeitig gibt es aber auch einen Wildwuchs, wer was wie aktualisiert – vom Betriebssystem-Dienst bis hin zum Browser-AddOn was sich selbst aktualisiert.

Für kleine Betriebe eignet sich oft AutoUpdate in Kombination mit einem regelmäßigen Scan durch einen (evtl. externen) Administrator. Für Installationen ab 200 Clients kann sich eine Lösung lohnen, die auch PCs provisionieren kann.

Viele der Anbieter von Client-Lösungen funktionieren prinzipiell auch für Server. Alle Programme finden Schwachstellen, nicht alle können die gefundenen Probleme auch beseitigen. Sowohl die Scan-Leistung als auch die Möglichkeiten zur Aktualisierung sind sehr unterschiedlich bezüglich Leistungsfähigkeit und Flexibilität.

IT.shavlik.com

■ Vorteile

- keine Infrastruktur nötig (Cloud-based)
- kostengünstiger Einstieg: 10 IPs kostenlos patchen
- Benutzer kann Reboot verzögern

■ Nachteile

- keine Grundinstallation / Provisionierung
- Probleme mit Firefox+.NET sowie mit Windows 7 HP
- nur für Windows (Clients und Server)

Shavlik bietet neben seinen klassischen Produkten auch eine SaaS-Lösung an.

- 10 IP+100 Scan/Monat kostenlos
25 IPs + 250 Scans/Monat = 250 US\$/Jahr
- Administrator kann alle Updates auf allen gescannten Rechnern auf ein Mal starten (was bei vielen Updates wohl überlegt sein will).

Shavlik NetChk Protect

■ Vorteile

- ohne Agenten verwendbar (über IPC\$)
- Office–Master–Installation (für Niederlassungen)
- enthält auch Viren–/Spyware–Scanner

■ Nachteile

- ab 40 US\$/Jahr/PC bzw. 80 US\$/Jahr/Server
- kleinere Installationsprobleme
- nur für Windows (Clients und Server)

NetChk hat in mehreren unabhängigen Testberichten sehr gut oder sogar als Testsieger abgeschnitten.

Das klassische Produkt von Shavlick ist auch als Bundle mit IT.shavlik.com oder mit dem hauseigenen Konfigurationsmanagement–Tool möglich.

matrix42 Empirum

■ Vorteile

- komplette Provisionierung neuer Systeme, automatisierbar (abhängig vom Admin-Know-How)
- sehr flexibel, sinnvoll ab 200 PCs

■ Nachteile

- hoher Aufwand für Schulung der Administratoren, Serverpflege und Paketieren von Software/Updates
- Lizenzkosten umgerechnet pro Client >100€/Jahr

Empirum ist eine Suite zum Software- und Konfigurationsmanagement, welche auch Patchmanagement abdeckt. Der Hersteller sitzt in Deutschland, was für Support und Haftung von Vorteil ist. Da Server und Clients separat zu lizenzieren sind und der Administrations-, Paketierungs- und Schulungsaufwand erheblich ist, lohnt sich der Einsatz erst wenn regelmäßig neue Arbeitsplätze provisioniert werden müssen.

OCS Inventory NG

■ Vorteile

- Open Source
- unterstützt viele Betriebssysteme
- inklusive Inventarisierung

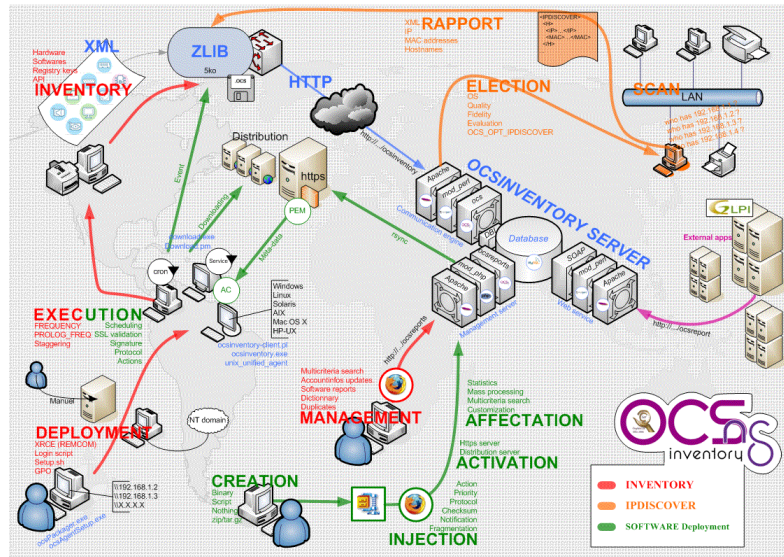
■ Nachteile

- komplex!, mehrere Server nötig
- agentenbasiert

OCS Inventory NG ist eine Open-Source Patchmanagement-Suite. Sie kann viele Anforderungen abdecken, ist aber dementsprechend komplex.

Der Autor hat keine Erfahrungen mit dieser Software, sie war aber nach seiner Einschätzung (außer den von den Distributionen mitgelieferten Werkzeugen) die einzige ernst zu nehmende Alternative im Open-Source-Bereich.

Flexibilität führt zu Komplexität



Diese Strukturübersicht von der OCS Inventory NG Webseite macht deutlich, wie Flexibilität durch Komplexität erkaufte wird.

Ausblick

- **SaaS/Cloud Computing**
 - Outsourcing der IT inkl. Patchmanagement
- **Secunia 2010: kostenloses Patchmanagement ?!**
- **AutoUpdate**
 - Google Chrome, Adobe Reader: Auswirkungen?
- **Microsoft CoApp: OpenSource@Microsoft**
- **heise Wunschzettel: Update4AllApps**

- **Mobile Devices/Smartphone**

- Secunia-Ankündigung:
<http://www.v3.co.uk/v3/news/2258725/rsa-2010-secunia-offer-free>
- Heise-Wunschzettel für Windows 7:
<http://heise.de/-270132>
- Nicht zwangsweise sicherheitsrelevant, aber nett: das Microsoft FixIt-Center:
<http://heise.de/-979890>

Fazit

- Updates sind mindestens genauso wichtig wie Firewall und Virens Scanner
- Legen Sie Ihren Fokus nicht aufs Betriebssystem: Plug-Ins, Office und Drittanbieter-Software sind genauso wichtig

VIELEN DANK FÜR IHR INTERESSE

Bleiben Sie am Ball – ich wüncbe allen Lesern ein gutes Gelingen!

- heise Security News vom 16.04.2010:
Kriminelle versuchen ungepatchte Reader-Lücke auszunutzen
<http://heise.de/-979037>



Betrieblicher Datenschutzbeauftragter nach BDSG
Unabhängige Datenschutz- und IT-Sicherheitsaudits
Beratung & Schulung für Führungskräfte und IT-Personal
Verschlüsselungskonzepte für Email und mobile Daten

Lösungen, die zu Ihrem Unternehmen passen

DATEN SCHUTZ
individuell

DATEN SCHUTZ *individuell*

20 Jahre Erfahrung in IT-Sicherheit und 10 Jahre Erfahrung im Datenschutz.

Olav Seyfarth ist bei mehreren mittelständischen Unternehmen in Südbaden zum betrieblichen Datenschutzbeauftragten benannt. Deren Ziel ist es vor allem, gesetzliche Vorschriften einzuhalten. Wie dies am Besten erreicht werden kann, hängt jedoch stark vom Unternehmen ab. Individuell bedeutet, technische, organisatorische und personelle Aspekte aufeinander abzustimmen und so eine optimale Lösung zu finden.

Alternativ zur Bestellung als Sicherheits- oder Datenschutzbeauftragter bietet Datenschutz individuell an, interne Datenschutzbeauftragte zu coachen oder Schulungen zu übernehmen. Mit einem Audit für Datenschutz und IT-Sicherheit („zweite Meinung“) können Geschäftsführer und Vorstände Ihrer Kontrollpflicht nachkommen.

Olav Seyfarth ist Diplom-Informatiker (FH), Certified Information Security Officer (ISTA), zertifizierter Datenschutzbeauftragter (TAR) und Fachkraft für Arbeitssicherheit (BG Druck). Er hat in weltweit tätigen mittelständischen Unternehmen Stabsstellen zur Informationssicherheit aufgebaut.

Weitere Informationen finden Sie auf <http://www.datenschutz-individuell.de/>