

Securing UNIX-Systems

Wie sichere ich mein System ab?

AmP

Chaos Computer Club Cologne e.V.
<http://koeln.ccc.de>

OpenChaos Januar
31.01.2008



Gliederung

- 1 Allgemeines
Warum?
Systeme
- 2 Angriffsmöglichkeiten
Allgemein
Webservices
- 3 Hardening
Applications
User-Land
Kernel-Level
Remote Sicherheit
Netzwerkweite Sicherheit
- 4 Abschluss
Ausblick



Gliederung

- 1 Allgemeines
 - Warum?
 - Systeme
- 2 Angriffsmöglichkeiten
 - Allgemein
 - Webservices
- 3 Hardening
 - Applications
 - User-Land
 - Kernel-Level
 - Remote Sicherheit
 - Netzwerkweite Sicherheit
- 4 Abschluss
 - Ausblick



Warum UNIX?

- Starke Verbreitung
- Kritische Bereiche
- Teilweise unerfahrene Admins
- Remote gut administrierbar
- Einfache Möglichkeiten für weitere Angriffe
- Desktop-Systeme (Ubuntu ...)
- Testserver-Problematik
- Alternativen?
- Frontends (Confixx, Webmin, Plesk)



Gliederung

- 1 Allgemeines
Warum?
Systeme
- 2 Angriffsmöglichkeiten
Allgemein
Webservices
- 3 Hardening
Applications
User-Land
Kernel-Level
Remote Sicherheit
Netzwerkweite Sicherheit
- 4 Abschluss
Ausblick



Ziele und Möglichkeiten

- DDoS
- Buffer Overflows
- Physikalische Angriffe (Notebooks!)
- Bewusste Manipulation
- Privilegien Eskalation
- Race Condition
- Datenmanipulation
- Remote root-expl0its
- Datenflut (Mailinglisten usw.)
- Netzwerkfreigaben



Webservices

- (SQL-) Injection
- XSS
- Directory Traversal
- Happy Expl0iting (CMS, Boards, Blogs)
- Infos sammeln / Informationsleaks
- Session-Management
- Bypassing Client Side Controls
- Shell providing

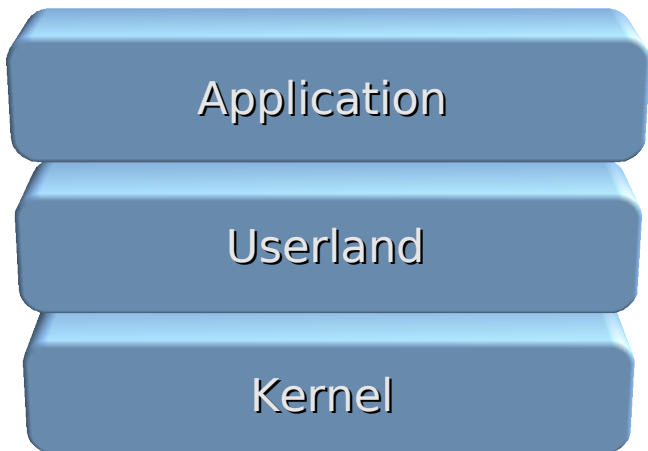


Gliederung

- 1 Allgemeines
Warum?
Systeme
- 2 Angriffsmöglichkeiten
Allgemein
Webservices
- 3 Hardening**
Applications
User-Land
Kernel-Level
Remote Sicherheit
Netzwerkweite Sicherheit
- 4 Abschluss
Ausblick



Hardening?



Applications

- Security-Erweiterungen
 - Server-Erweiterungen
 - Compiler-Erweiterungen
- Security Checking Tools (NIST, BSI, usw.)
- Zentrales Logging und Auswertung
- Access Control
 - Von wo? Wie?
 - Mit welcher Login-Art darf zugegriffen werden
- Authentizität und Integrität der Software
- Hardening VOR dem Launch der Software
- Default-Werte und Default-Accounts
- Komplexe Dienstekennwörter
- Chroot / Jail
- Dienste-Continuity



User-Land

- Nutzer-Konten
- Frontends (Confixx, Webmin, Plesk) wirklich nötig?
- Überwachung der Funktionsfähigkeit
- Verfügbarkeit Sicherstellen
- Updates
- Firewalling
 - Netzwerkweit
 - Lokal
- Security Scans
- Mount Point Sicherheit (nodev, noexec, usw)
- User brauchen keinen compiler
- Dateisystemverschlüsselung (swap!)
- Regelmässige (Config-)Reviews



- Passwortauthentifizierung vermeiden
 - Public-/Private-Key
 - X.509 Certificate
 - One-Time-Passwords
 - Arten der Passwort-Speicherung
 - Zentrales Account-Management
- Access Control Lists
 - Discretionary vermeiden (oder restriktive Permissions)
 - Mandatory oder Role-based
- Virtualisierung
- Tools
 - bastille
 - AIDE
 - LIDS
 - OSSEC
 - chrootkit / rootkithunter



Kernel-Level

- Speicherbereiche Schützen
- Syscall Sandboxing
- Panic-Handling
- Virtualisierung
- Partitionierung
- Tools
 - PaX oder W^X
 - Grsecurity
 - SELinux
 - TrustedBSD
 - AppArmor (Entwickelt das noch wer?)
 - Trusted Extensions (TrustedSlowlaris)



Remote Sicherheit

- Sicherer Zugriff (SSH ftw)
- tcpwrapper (hosts.allow und hosts.deny)
- Lokale firewalls
 - pf oder ipfw für BSD
 - Netfilter/iptables für Linux
 - IPFilter für den Rest
 - -> Konzept !?!
- verschiedene (virtuelle) Hosts
- Banners abschalten oder reduzieren
- UPDATES!
- Offensive Security
 - BackTrac (nikto, raccess, nessus)
 - Benchmarking



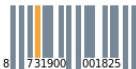
Netzwerkweite Sicherheit

- Mehrstufiges Firewalling
- Zentrales Logging
- Überwachung
 - nagios
 - bb
 - cacti
 - MRTG
 - munin
- Benachrichtigung
- Worst-Case-Szenarios
- Netzwerktrennung (Abteilungen, DMZ, Server, usw.)



Gliederung

- 1 Allgemeines
Warum?
Systeme
- 2 Angriffsmöglichkeiten
Allgemein
Webservices
- 3 Hardening
Applications
User-Land
Kernel-Level
Remote Sicherheit
Netzwerkweite Sicherheit
- 4 **Abschluss**
Ausblick



- Host-Security immer wichtiger (IPv6)
- Automatische Angriffe immer häufiger
- Leichter (anonymer) Netzwerkzugriff
- Mehr Server
- Mehr Leistungsstarke Ziele



Quellenverzeichnis

Securing Debian <http://www.debian.org/doc/manuals/securing-debian-howto>

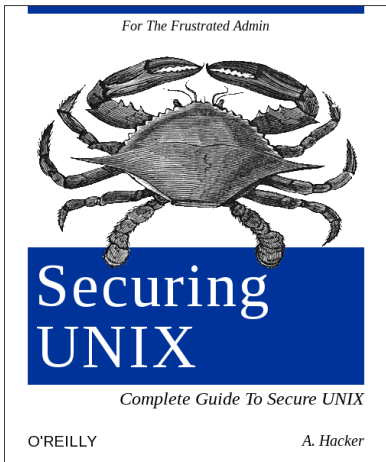
Wikipedia Gebt doch einfach mal die hier vorgestellten Techniken ein

Diverse Projekt-Pages

Vendors Fast alle Hersteller/Projekte haben HowTos



Fragen



Fragen Diskussion

