Absicherung von Linux-Rechnern mit grsecurity

Brandenburger Linux Infotag, 23. April 2005

Wilhelm Dolle, Director Information Technology
interActive Systems GmbH

Agenda

- Grundlagen und Historie von grsecurity
- Installation
- Buffer-Overflows
- Role-Based Access Control
- Randomisierung im TCP/IP-Stack
- Fazit

Die Herausforderung

- Software enthält (immer) Fehler
  - Design, Implementierung, Konfiguration
  - Fehlersuche und -behebung ist sehr aufwändig
  - Fehler können nicht vollständig ausgeschlossen werden
- Rechner die Dienste anbieten sind (immer) angreifbar
  - Mehr Dienste erzeugen höhere Gefährdung
- Angriffs-Tools werden immer leichter zugänglich und einfacher zu bedienen
  - Viren, Würmer, Rootkits, trojanische Pferde, ...
- Lösungsansatz A: Software fehlerfrei machen
- Lösungsansatz B: Ausnutzen der Fehler verhindern

Historie von grsecurity

- Start des Projektes in 2001
- Autor: Brad Spengler
- Lizenz: GNU GPL
- Ursprünglich nur ein Port von OpenWall auf Linux
- Erste Veröffentlichung für Kernel 2.4.1
- Aktuelle Version: 2.1.5 (für Kernel 2.4.30 / 2.6.11.7)

Philosophie von grsecurity

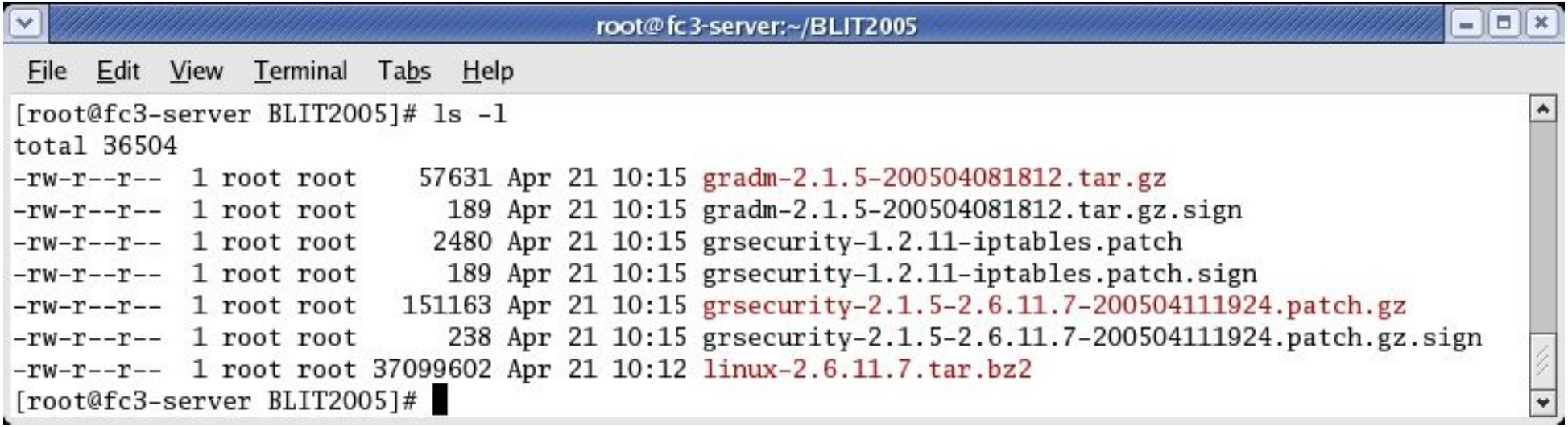
- Sicherheit kann nicht auf einer einzigen Ebene erreicht werden
- Zusätzliche Sicherheit muss benutzerfeundlich sein um wirken zu können
- Jede Software auf dem System muss zu schützen sein (nicht nur die aus der Distribution selber)
- Der Mensch ist das schwächste Glied in der Sicherheitskette
- Strategie: “Detection, Prevention and Containment”

Haupt-Features von grsecurity

- Verhindern des Ausnutzens von Buffer-Overflows (Buffer Overflow exploitation prevention - PaX)
- Role-Based Access Control (RBAC)
- Zusätzliche Randomisierung im TCP/IP-Stack und bei den Prozess-IDs
- Eingeschränkte Sichtbarkeit von Prozessen
- Change root (chroot) hardening
- Schutz vor Race-Conditions (hauptsächlich /tmp)
- Umfangreiches Auditing

Installation von grsecurity I

- **Was wird benötigt?**
  - **(aktueller) Linux-Kernel von kernel.org**
  - **grsecurity-Kernel-Patch**
  - **gradm**
  - **Prüfsummen**
  - **Compiler :-)**



Installation von grsecurity II

- **Kernel entpacken**
- **Patch einspielen**

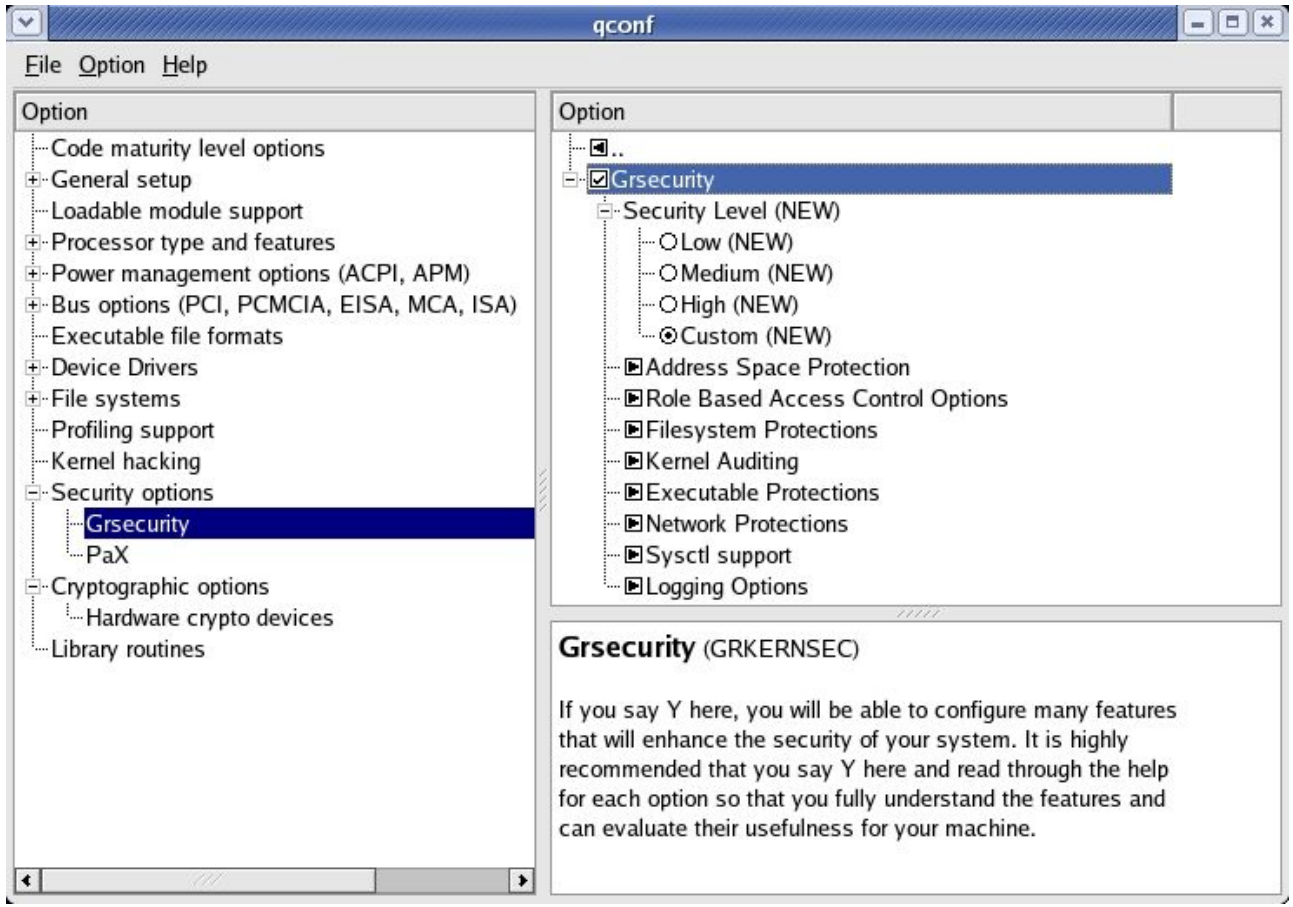
root@fc3-server:/usr/src

File Edit View Terminal Tabs Help

```
[root@fc3-server src]# ls -l
total 36432
-rw-r--r--  1 root root   151163 Apr 21 10:18 grsecurity-2.1.5-2.6.11.7-200504111924.patch.gz
-rw-r--r--  1 root root 37099602 Apr 21 10:18 linux-2.6.11.7.tar.bz2
drwxr-xr-x  7 root root     4096 Jan  9 23:28 redhat
[root@fc3-server src]# tar jxf linux-2.6.11.7.tar.bz2
[root@fc3-server src]# gunzip grsecurity-2.1.5-2.6.11.7-200504111924.patch.gz
[root@fc3-server src]# patch -p0 < grsecurity-2.1.5-2.6.11.7-200504111924.patch
patching file linux-2.6.11.7/Makefile
patching file linux-2.6.11.7/arch/alpha/kernel/osf_sys.c
patching file linux-2.6.11.7/arch/alpha/kernel/ptrace.c
patching file linux-2.6.11.7/arch/alpha/mm/fault.c
patching file linux-2.6.11.7/arch/arm/mm/fault.c
patching file linux-2.6.11.7/arch/arm/mm/mmap.c
patching file linux-2.6.11.7/arch/arm26/mm/fault.c
patching file linux-2.6.11.7/arch/cris/mm/fault.c
patching file linux-2.6.11.7/arch/i386/Kconfig
patching file linux-2.6.11.7/arch/i386/kernel/apm.c
```

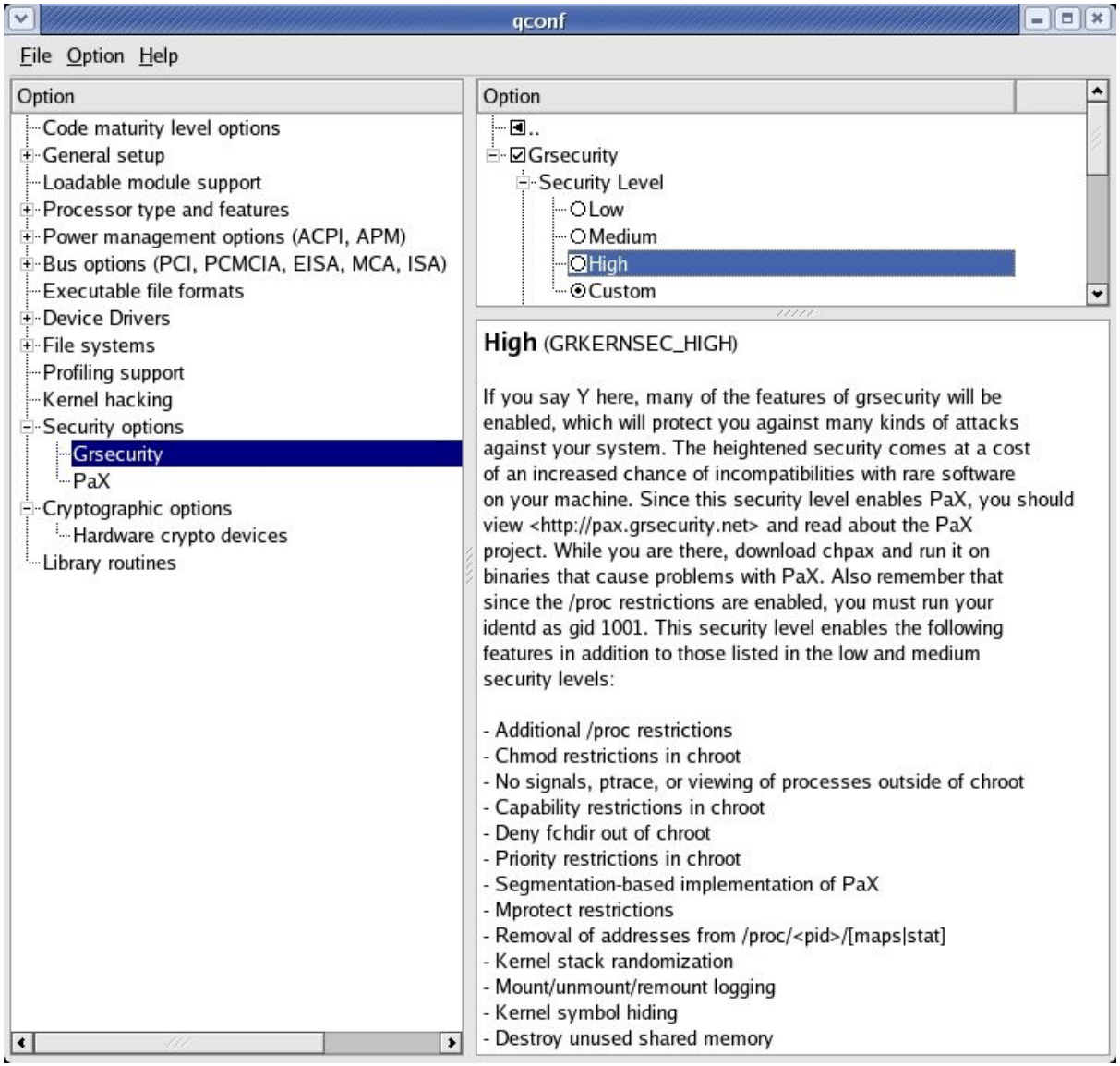
Installation von grsecurity III

- **Kernel compilieren und aktivieren**
  - **Hilfe auch unter grsecurity.net/confighelp.php**
  - **Generische Konfiguration grsecurity.net/generic-config**
  - **Hier: make xconfig / Fedora Core 3 / Kernel 2.6.11.7**

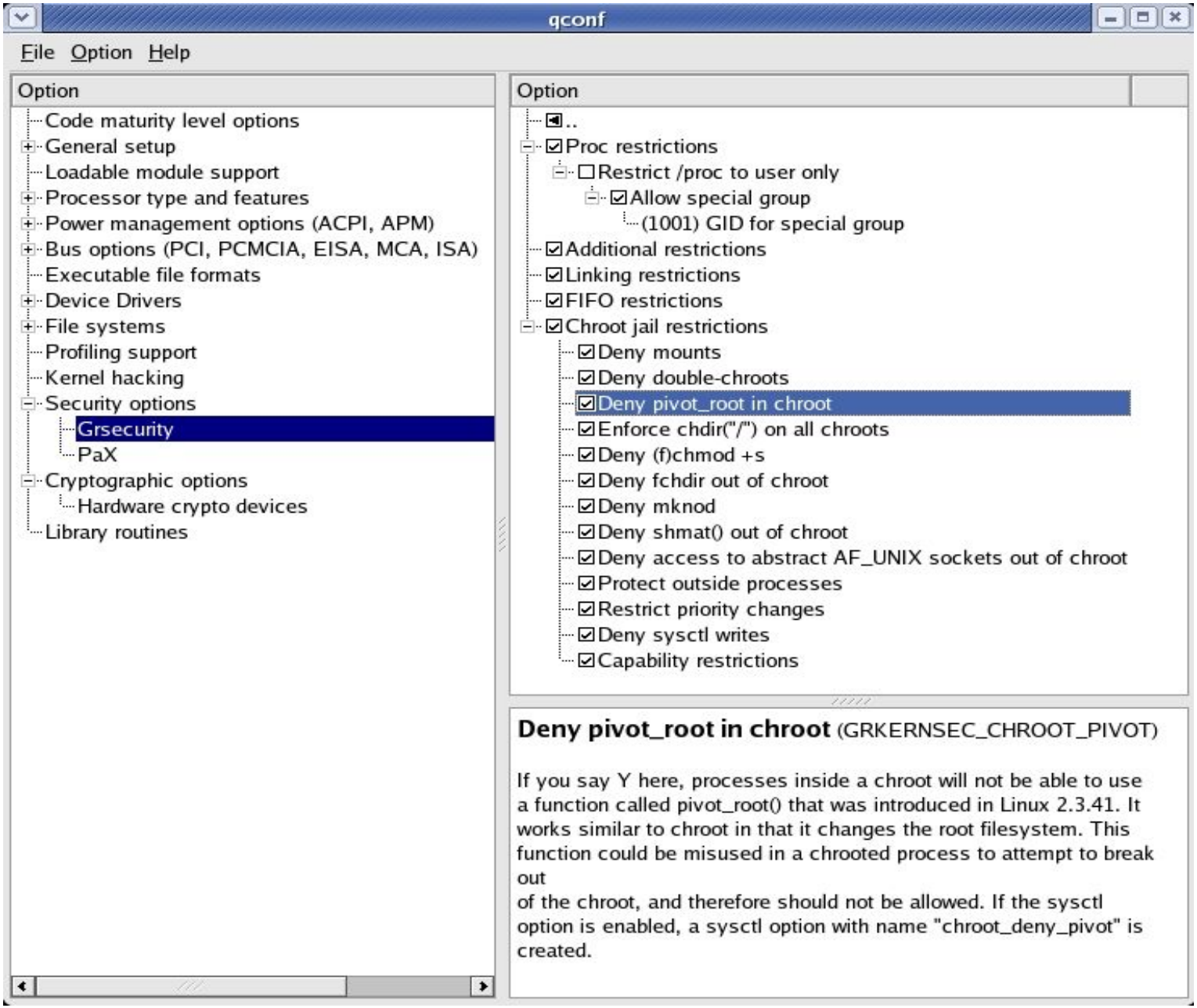


Konfiguration I

- **Mittels Kernel-Konfiguration, gradm oder über /etc/sysctl.conf**



Konfiguration II



Buffer-Overflows I

- **SANS Top 20 Internet Security Vulnerabilities http://www.sans.org/top20**

Top Vulnerabilities to Windows Systems

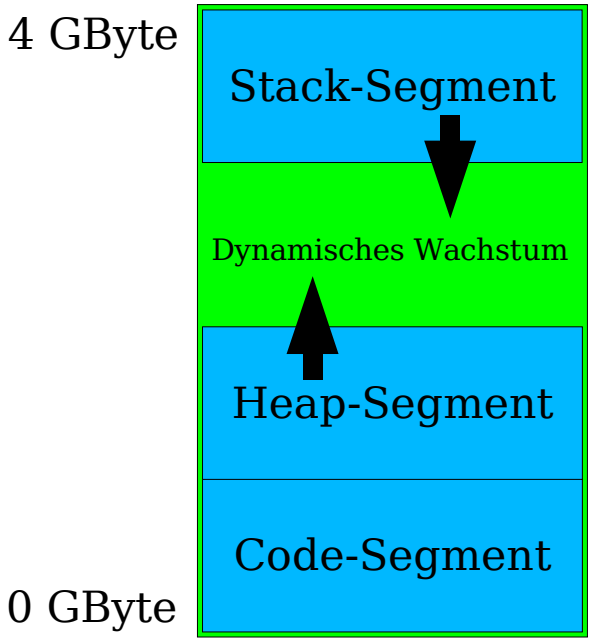
- W1 Web Servers & Services
- W2 Workstation Service
- W3 Windows Remote Access Services
- W4 Microsoft SQL Server (MSSQL)
- W5 Windows Authentication
- W6 Web Browsers
- W7 File-Sharing Applications
- W8 LSAS Exposures
- W9 Mail Client
- W10 Instant Messaging

Top Vulnerabilities to UNIX Systems

- U1 BIND Domain Name System
- U2 Web Server
- U3 Authentication
- U4 Version Control Systems
- U5 Mail Transport Service
- U6 Simple Network Management Protocol (SNMP)
- U7 Open Secure Sockets Layer (SSL)
- U8 Misconfiguration of Enterprise Services NIS/NFS
- U9 Databases
- U10 Kernel

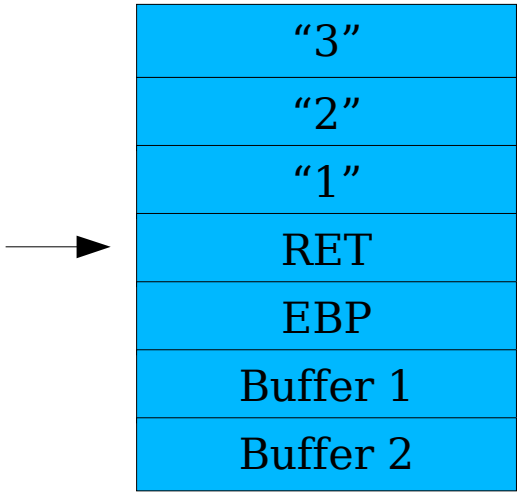
Buffer-Overflows II

- **Prozessspeicher Layout**
  - **Code-Segment (Text Segment) – hier liegt der Programmcode selber**
  - **Heap-Segment (Daten Segment) – Variablen (extern, static), Felder und Tabellen des Prozesses**
  - **Stack-Segment – dynamische Variablen, übergebene Paramenter und Rücksprungadressen von Funktionen**



Buffer-Overflows III

- Stack-Aufbau
  - Last-in-first-out
  - ESP (extended stack pointer) – oberstes Element des Stacks
  - EBP (frame pointer / extended base pointer) – unterstes Element des aktuellen Stacks
  - EIP (extended intruction pointer) – Speicheradresse der nächsten Instruktion



Buffer-Overflows IV

- Verhindern des Ausnutzens von Buffer-Overflows
  - Nicht ausführbarer Stack
  - Canaries auf dem Stack
  - Kritische Funktionen durch spezielle Bibliotheken ersetzen
- PaX – SEGMEXEC
  - Nicht ausführbare User-Pages durch die segmentation logic der IA-32 (Intel x86) Architektur und virtual memory area mirroring
- PaX – PAGEEXEC
  - Auf Plattformen die ein “executable bit” hardwaremäßig unterstützen, wie Alpha, PPC, Sparc, Sparc64, amd64 und ia64
- PaX – ASLR (address space layout randomization)

Role-Based Access Control I

- Einführen von Rollen die eine Aufgabe des Benutzers in einer Organisation beschreiben (z.B. Revisor, Backup-Operator, Email-Administrator, System-Administrator)
- Aufteilen der mächtigen “root”-Berechtigungen in einzelne Teile
- RBAC-Systeme übernehmen Teile von MAC-Ansätzen (Mandatory Access Control) und weisen Rechte auf der Ebene von Subjekten und Objekten zu

Role-Based Access Control II

- “User”-Rollen (gebunden an User-ID)
- “Group”-Rollen (gebunden an Group-ID)
- “Special”-Rollen (frei zuzuweisen)

- Grsecurity vergibt niemals mehr Rechte als das Linux-System es tun würde (z.B. Einschränken von User-ID 0)

- IP-basierte Rollen

- Rollenhierarchie: special, user, group, default

- Rollenübergangstabellen (authentifiziert / nicht authentifiziert) -> zusätzlicher Schutz

- Intelligentes Lernsystem (über gradm)

Randomisierung I

- **Randomisierung bei TCP-Quell-Ports**

wd@T42p:~

```
[wd@T42p ~]$ uname -a
Linux T42p.wdolle.de 2.6.9-1.681_FC3 #1 Thu Nov 18 15:10:10 EST 2004 i686 i686 i386 GNU/Linux
[wd@T42p ~]$ netstat -nt
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
tcp        0      0 192.168.53.1:32771          192.168.53.150:22           ESTABLISHED
tcp        0      0 192.168.53.1:32773          192.168.53.150:22           ESTABLISHED
tcp        0      0 192.168.53.1:32772          192.168.53.150:22           ESTABLISHED
tcp        0      0 192.168.53.1:32775          192.168.53.150:22           ESTABLISHED
tcp        0      0 192.168.53.1:32774          192.168.53.150:22           ESTABLISHED
tcp        0      0 192.168.53.1:32777          192.168.53.150:22           ESTABLISHED
tcp        0      0 192.168.53.1:32776          192.168.53.150:22           ESTABLISHED
[wd@T42p ~]$
```

wd@bgpd:~

```
[wd@bgpd wd]$ netstat -atn | grep 127.0.0.1
tcp        0      0 127.0.0.1:6868          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:6969          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:22            127.0.0.1:59008         ESTABLISHED
tcp        0      0 127.0.0.1:46953         127.0.0.1:22            TIME_WAIT
tcp        0      0 127.0.0.1:38280         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:43525         ESTABLISHED
tcp        0      0 127.0.0.1:53356         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:45158         ESTABLISHED
tcp        0      0 127.0.0.1:45158         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:45150         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:38280         ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:42504         ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:53356         ESTABLISHED
tcp        0      0 127.0.0.1:59008         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:42504         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:43525         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:51999         127.0.0.1:22            ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:45150         ESTABLISHED
tcp        0      0 127.0.0.1:22            127.0.0.1:51999         ESTABLISHED
[wd@bgpd wd]$
```

Linux: von 32.768 bis 61.000 (jeweils um 1 erhöhen)

Randomisierung II

- TCP-Reset-Angriff auf eine TCP-Verbindung über typische T-DSL-Verbindung (128 Kbit/s Upstream)
  - "Erraten" werden muss die Sequenznummer (32 Bit)
  - Größe des Reset-Paketes (IP- und TCP-Header) 40 Byte
  - Typische Window-Größe bei Linux 2.4 / 2.6 ist 5840 Bit
  - Quellport bekannt (kann bestimmt werden):

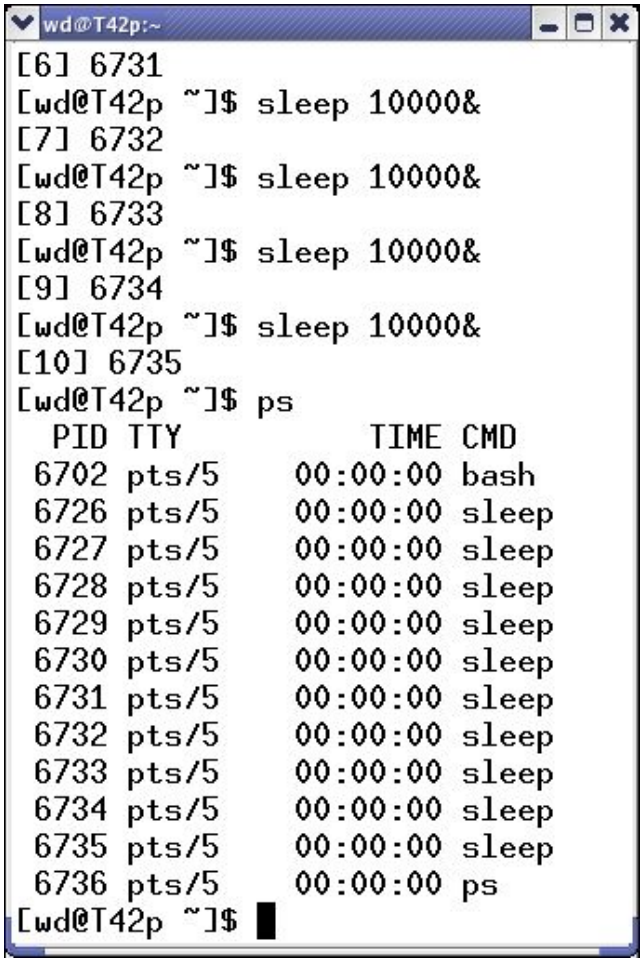
    4.294.967/5.840*320/128.000/2 = 15:19 Minuten
  - Quellport unbekannt:

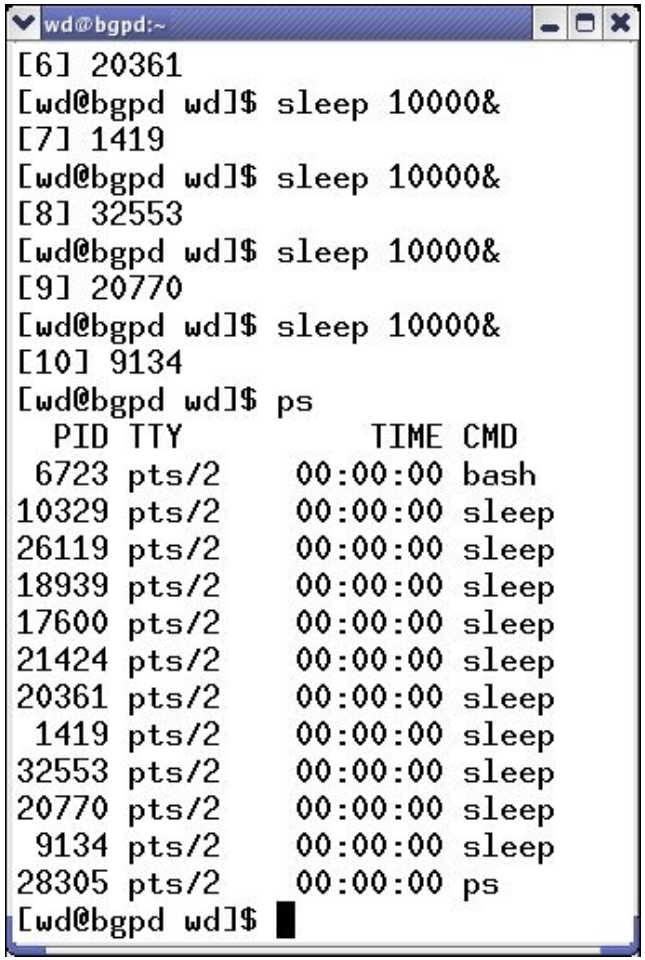
    919s * 65.535 = 697 Tage

- Siehe auch letzte Microsoft Window-Size-Patches Mitte April 2005 (65.535 auf 17.520)

Randomisierung III

- **Zufällige Prozess-IDs verhindern unter anderem das “Erraten” der Namen von temporären Dateien (Gefahr von Race Conditions)**





Fazit

- **Im Vergleich zur Entwicklung fehlerfreier Software ist das Verhindern des Ausnutzens der Schwachstellen einfacher und kostengünstiger (trotzdem ist fehlerarme Software anzustreben)**
- **Wirkt auch gegen unbekannte Bugs**
- **Kann “Patchorgien” verhindern**

Fragen?

Vielen Dank für die Aufmerksamkeit

Folien in ein paar Tagen unter: http://www.dolle.net

Wilhelm Dolle, CISA, CISSP, BSI IT-Grundschutz-Auditor
Director Information Technology

iAS interActive Systems
Dieffenbachstrasse 33c
D-10967 Berlin

phone +49-(0)30-69004-100
fax +49-(0)30-69004-101
mail wilhelm.dolle@interActive-Systems.de
web http://www.interActive-Systems.de