Patch Management

How to roll out and track security fixes to your systems

Klaus Möller

WP8-T1

Webinar, 10th of June 2021

Public

www.geant.org
What we will cover today

• Process
  - NIST SP 800-40r3 “Guide to Enterprise Patch Management Technologies”
  - NIST SP 800-40 version 2 “Creating a Patch and Vulnerability Management Program”

• Tools
  - Linux: zypper, yum, apt
  - Windows: SUMo, PatchMyPC
Definition

• Patch: “is an additional piece of code developed to address problems (commonly called ‘bugs’) in software.” (NIST SP 800-40r2)

• Variants/other names
  – Hotfix
  – Point release
  – Program temporary fix (PTF)
  – Security patch
  – Service Pack
  – Unofficial patch
  – Monkey patch
Patch Management

- Patch Management is the process for
  - identifying
  - acquiring
  - installing, and
  - verifying
- patches for products and systems (NIST SP 800-40r3)
Identification
Identifying

- Systems that have Software running that needs to be patched
  - From inventory/asset management
  - From network scanning
  - From passive discovery
- Software, that has vulnerabilities or is misconfigured
  - Installation database on systems
  - Local scanning of process lists, file systems
- Vulnerabilities, that have impacts
  - Vulnerability databases, security advisories, etc.
- Patches/Remediations, available for vulnerabilities
  - Security advisories, vendor update channels, ...
Linux Version

- **File: `/etc/os-release`**
  - Machine readable: `CPE_NAME`
  - Human readable: `PRETTY_NAME`

```bash
NAME="CentOS Linux"
VERSION="7 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="7"
PRETTY_NAME="CentOS Linux 7 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:7"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"

NAME="openSUSE Leap"
VERSION="15.2"
ID="opensuse-leap"
ID_LIKE="suse opensuse"
VERSION_ID="15.2"
PRETTY_NAME="openSUSE Leap 15.2"
ANSI_COLOR="0;32"
CPE_NAME="cpe:/o:opensuse:leap:15.2"
HOME_URL="https://www.opensuse.org/"
```

```bash
PRETTY_NAME="Kali GNU/Linux Rolling"
NAME="Kali GNU/Linux"
ID="kali"
VERSION="2021.2"
VERSION_ID="2021.2"
VERSION_CODENAME="kali-rolling"
ID_LIKE=\"debian\"
ANSI_COLOR=\"1;31\"
HOME_URL=\"https://www.kali.org/\"
SUPPORT_URL=\"https://forums.kali.org/\"
BUG_REPORT_URL=\"https://bugs.kali.org/\"
```
Installed Software on Linux

- **Installed Software:**
  - `rpm -qa`  # RedHat, Fedora, CentOS, SuSE, ...
    - Berkeley DB files in `/var/lib/rpm/`
  - `dpkg -l`  # Debian, Ubuntu, Kali, ...
    - ASCII files in `/var/lib/dpkg/`

```
$ dpkg -l
Desired=Unknown/Install/Remove/Purge/Hold
 | Status=Not/Inst/Conf-files/Unpacked/half-conf/Half-inst/trig-aWait/Trig-pend
 | Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
 ||/ Name               Version               Architecture Description
+++-===================-=====================-============-============
ii  0trace             0.01-3kali2           amd64        traceroute tool that can run with...
ii  aapt               1:10.0.0+r36-3        amd64        Android Asset Packaging Tool
ii  accountsservice    0.6.55-3              amd64        query and manipulate user account ...
ii  acl                2.2.53-10             amd64        access control list - utilities
ii  adduser            3.118                 all          add and remove users and groups
...```
## Linux: Pending Patches/Updates

- **List pending patches/updates**
  - `zypper list-patches`  # openSuse & derivates
  - `yum updateinfo`  # RedHat & derivates (Fedora, CentOS, ...)
  - `apt list --upgradable`  # Debian & derivates

```bash
> zypper list-patches --issue
```

The following matches in issue numbers have been found:

<table>
<thead>
<tr>
<th>Issue</th>
<th>No.</th>
<th>Patch</th>
<th>Category</th>
<th>Severity</th>
<th>Interactive</th>
<th>Status</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>bugzilla</td>
<td>1170160</td>
<td>openSUSE-2021-463</td>
<td>recommended</td>
<td>low</td>
<td>---</td>
<td>needed</td>
<td>Recommended update for hwdata</td>
</tr>
<tr>
<td>bugzilla</td>
<td>1182482</td>
<td>openSUSE-2021-463</td>
<td>recommended</td>
<td>low</td>
<td>---</td>
<td>needed</td>
<td>Recommended update for hwdata</td>
</tr>
<tr>
<td>bugzilla</td>
<td>1172442</td>
<td>openSUSE-2021-468</td>
<td>security</td>
<td>important</td>
<td>---</td>
<td>needed</td>
<td>Security update for nghttp2</td>
</tr>
<tr>
<td>bugzilla</td>
<td>1181358</td>
<td>openSUSE-2021-468</td>
<td>security</td>
<td>important</td>
<td>---</td>
<td>needed</td>
<td>Security update for nghttp2</td>
</tr>
<tr>
<td>cve</td>
<td>CVE-2020-11080</td>
<td>openSUSE-2021-468</td>
<td>security</td>
<td>important</td>
<td>---</td>
<td>needed</td>
<td>Security update for the Linux Kernel</td>
</tr>
<tr>
<td>bugzilla</td>
<td>1180597</td>
<td>openSUSE-2021-506</td>
<td>recommended</td>
<td>moderate</td>
<td>restart</td>
<td>needed</td>
<td>Recommended update for PackageKit</td>
</tr>
<tr>
<td>cve</td>
<td>CVE-2021-29155</td>
<td>openSUSE-2021-716</td>
<td>security</td>
<td>important</td>
<td>reboot</td>
<td>needed</td>
<td>Security update for the Linux Kernel</td>
</tr>
<tr>
<td>cve</td>
<td>CVE-2021-29650</td>
<td>openSUSE-2021-716</td>
<td>security</td>
<td>important</td>
<td>reboot</td>
<td>needed</td>
<td>Security update for the Linux Kernel</td>
</tr>
<tr>
<td>bugzilla</td>
<td>1183797</td>
<td>openSUSE-2021-744</td>
<td>optional</td>
<td>low</td>
<td>---</td>
<td>optional</td>
<td>Optional update for sed</td>
</tr>
</tbody>
</table>
Windows Version

- Workstation & Server
  - (NT 4, 2000, ..., 10)
- Keys
  - DisplayVersion
  - ProductName
  - ReleaseID
- Scripting
  - CLI
    - `reg`
  - PowerShell (WMI)
    - `(Get-CimInstance Win32_OperatingSystem)`
    - `Get-ComputerInfo (PS ≥ version 5)`
Installed Software on Windows

- **Two GUIs**
  - Windows 10: Settings → Apps → Apps and Features
  - System → Programs → Programs & Features

- **Scripting**
  - PowerShell

```
PS > get-WmiObject -Class Win32_product | Select Name, ... | Format-Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Vendor</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Python 3.9.2 Add to Path (64-bit)</td>
<td>Python Software Foundation</td>
<td>3.9.2150.0</td>
</tr>
<tr>
<td>Python 3.9.2 Utility Scripts (64-bit)</td>
<td>Python Software Foundation</td>
<td>3.9.2150.0</td>
</tr>
<tr>
<td>Python 3.9.2 Core Interpreter (64-bit)</td>
<td>Python Software Foundation</td>
<td>3.9.2150.0</td>
</tr>
<tr>
<td>KeePassXC</td>
<td>KeePassXC Team</td>
<td>2.6.4</td>
</tr>
</tbody>
</table>
```
Acquisition
Acquiring Patches: Sources

• Vendor update server
  – Directly from the source without delay
  – Saves effort for deploying and maintaining local infrastructure

• Local (own) update server
  – Less traffic on the uplink
  – A means to include your own patches
  – Works when internet is down
  – Nobody can see what is actually being deployed/patched
  – More work, however

• Manually deploying patch media?
  – Do you have air-gapped systems?
  – You will, after a major breakdown!
  – What about applying patches to installation media (slipstreaming)?
Acquiring Patches: Tools

- **Microsoft**
  - System Management Server (SMS)
    - Systems Management Server Inventory Tool (ITMU) for Microsoft Update
    - System Center Configuration Manager (SCCM)
  - Windows Server Update Services (WSUS)

- **Linux**
  - Mirroring vendor repositories
    - RedHat: `reposync` + HTTP server
    - Debian: `apt-mirror` + HTTP server
    - SuSE: `rsync` + HTTP server
  - Linux vendor tools: RedHat Satellite, etc.

- Configure (all) local systems to use local update server
Installation
Installation

- How is the installation done?
  - Typically not a big deal, unless you have to compile it yourself
- Who does it?
  - What about systems nobody is responsible for
- Side effects
  - Interrupts/Downtime
    - What about hypervisors, container hosts?
- What if something breaks?
  - Best practice: Take filesystem snapshot to rollback if something goes wrong
    - Sometimes done automatically (Windows Update, SuSE YaST/snapper)
Snapshot Tools

- **Linux: snapper**
  - Requires snapshot-capable filesystem (BTRFS, ...), or volume snapshots (LVM)

```plaintext
> snapper list

<table>
<thead>
<tr>
<th>#</th>
<th>Type</th>
<th>Pre #</th>
<th>Date</th>
<th>User</th>
<th>Cleanup</th>
<th>Description</th>
<th>Userdata</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>single</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>current</td>
<td></td>
</tr>
<tr>
<td>1647</td>
<td>pre</td>
<td>1647</td>
<td>Mon May 17 08:42:08 2021</td>
<td>root</td>
<td>number</td>
<td>zypp(zypper)</td>
<td>important=yes</td>
</tr>
<tr>
<td>1648</td>
<td>post</td>
<td>1647</td>
<td>Mon May 17 08:45:18 2021</td>
<td>root</td>
<td>number</td>
<td>zypp(zypper)</td>
<td>important=yes</td>
</tr>
<tr>
<td>1674</td>
<td>pre</td>
<td>1674</td>
<td>Thu May 20 09:20:24 2021</td>
<td>root</td>
<td>number</td>
<td>zypp(zypper)</td>
<td>important=no</td>
</tr>
<tr>
<td>1691</td>
<td>pre</td>
<td>1691</td>
<td>Tue May 25 09:52:26 2021</td>
<td>root</td>
<td>number</td>
<td>zypp(zypper)</td>
<td>important=no</td>
</tr>
<tr>
<td>1692</td>
<td>post</td>
<td>1691</td>
<td>Tue May 25 09:55:23 2021</td>
<td>root</td>
<td>number</td>
<td>zypp(zypper)</td>
<td>important=no</td>
</tr>
<tr>
<td>1693</td>
<td>single</td>
<td></td>
<td>Tue May 25 10:00:07 2021</td>
<td>root</td>
<td>timeline</td>
<td>timeline</td>
<td></td>
</tr>
</tbody>
</table>

...```

- **Windows Volume Shadow Copy Service: VSSVC.exe**
  - Requires NTFS filesystem
  - Must be enabled/configured first
Update Mechanisms

• Software automatically updates itself
  – Browsers (Chrome), Mail Clients (Thunderbird), etc.

• OS management tool initiates patching
  – Windows Update, Linux PackageKit, ...

• User manually directs software to update itself
  – Sometimes by being notified by the software

• Third-party patch management application initiates patching
  – ManageEngine Patch Manager Plus, Atera, Syxsense, OPSI, ...

• Other tool initiates patching
  – Network access control, health check technologies, etc.

• Patch or new version is installed manually
**3rd Party Tools: Patch My PC**

- Freeware: Patch My PC Home Updater
  - Full list of supported 3rd party software directly visible
3rd Party Tools: Software Update Monitor (SUMo)

- Freeware: SUMo lite
- Companion tool: Driver Update Monitor (DUMo)

😊 Nag screen/icons
Challenges

- Timing (after the vulnerability is published, exploits happen)
  - When will the patch be released?
    - Risk of exploitation increases as time passes
    - Vendors follow their own schedule - sometimes weeks or months until patch is published
  - What to do in the meantime?
    - Workarounds, hotfixes, disabling services, etc.

- Prioritisation
  - Which patches to install when and where?
  - Side effects of patches
  - Time & resources for testing of patches

- Multiple Update Mechanisms
  - Blacklisting of patches becomes a challenge
  - Assuming that some other tool/admin might already be doing the update
Verification

www.geant.org
Verifying

- **Why?**
  - Things may break during patching: power outage, shutdown, connection loss, etc.
  - Rollback of partly-applied patches?
- **Even if the patch was successful, did it fix the vulnerability?**
  - Does the system need to reboot? ( Kernel patches, system libraries, ... )
  - Does the service need to restart?
  - What if we see further exploits happening?
- **What else could be affected?**
  - Performance, memory consumption, disk space
  - Incompatibilities, configuration changes
  - Patch itself might be broken, etc.
**Verification: Linux**

- List installed (security) patches
  - `zypper patches | grep 'security.*applied'` # openSuse, ...
  - `yum updateinfo list security installed` # RedHat, ...
  - `zgrep CVE /usr/share/doc/*/changelog.Debian.gz` # Debian, ...
  - Or package installer history files: `/var/log/apt/`, `/var/log/zypp/`, `/var/log/yum.log`
  - `pakiti` # Network-wide

<table>
<thead>
<tr>
<th>Repository</th>
<th>Name</th>
<th>Category</th>
<th>Severity</th>
<th>Interactive</th>
<th>Status</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Update Repository</td>
<td>openSUSE-2021-435</td>
<td>security</td>
<td>moderate</td>
<td>---</td>
<td>applied</td>
<td>Security update for python</td>
</tr>
<tr>
<td>Main Update Repository</td>
<td>openSUSE-2021-443</td>
<td>security</td>
<td>moderate</td>
<td>---</td>
<td>applied</td>
<td>Security update for privoxy</td>
</tr>
<tr>
<td>Main Update Repository</td>
<td>openSUSE-2021-59</td>
<td>security</td>
<td>moderate</td>
<td>restart</td>
<td>applied</td>
<td>Security update for libzypp, zypper</td>
</tr>
<tr>
<td>Main Update Repository</td>
<td>openSUSE-2021-6</td>
<td>security</td>
<td>moderate</td>
<td></td>
<td>applied</td>
<td>Security update for privoxy</td>
</tr>
<tr>
<td>Main Update Repository</td>
<td>openSUSE-2021-60</td>
<td>security</td>
<td>important</td>
<td>reboot</td>
<td>applied</td>
<td>Security update for the Linux Kernel</td>
</tr>
<tr>
<td>Main Update Repository</td>
<td>openSUSE-2021-89</td>
<td>security</td>
<td>important</td>
<td></td>
<td>applied</td>
<td>Security update for open-iscsi</td>
</tr>
</tbody>
</table>
Verification: Linux (cont.)

- Verifying installed software

Debian, Ubuntu, Kali, etc.

```
# dpkg -V
??5?????? c /etc/vpnc/default.conf
??5?????? c /etc/king-phisher/server_config.yml
??5?????? c /etc/snmp/snmpd.conf
??5?????? c /etc/redis/redis-openvas.conf
??5?????? c /etc/redis/redis.conf
??5?????? c /etc/ipsec.secrets
??5?????? c /etc/sudoers.d/kali-grant-root
??5?????? c /etc/openfortivpn/config
??5?????? c /etc/sudoers
```

File, content (checksum) only

RedHat, Fedora, CentOS, SuSE, etc.

```
# rpm -Va
S.5.....T. c /etc/default/useradd
..........T. c /etc/login.defs
`.M....UG.. g /run/knot
`.M....... g /run/cryptsetup
S.5.....T. c /etc/clamd.conf
S.5.....T. c /etc/freshclam.conf
`.M....... g /var/log/alternatives.log
```

- File size
- File content (checksum)
- File permissions
- File ownership (user/group)
- File modification times
Verification: Windows

- Windows Update
  - Update history
Wrapping Up
Testing Patches

• A broken patch can disrupt operations
  – If many or vital systems are affected
  – Personnel has to be assigned to do emergency clean-up
• So test all patches first?
  – Resources too limited for that
  – Availability of separate hardware for testing (non-production)
  – Takes time during which the risk increases further
• Alternatives
  – Test only for vital systems/services
  – Apply patches in smaller batches and look for things that go wrong
    • So you might stop the patch before being deployed to all systems
Patch Policy

- Develop one for your organisation
  - ISO 27001, GDPR require patch management
- Yes, takes time and effort, but
  - Issues get on the table - and are resolved, hopefully
  - Buy-in from management
    - Nobody should be fired for adhering to agreed-upon policies
  - Managing expectations
  - Everybody knows what to do
- Write down what you are doing already
  - So you have a starting point for discussion
  - What would you like to improve?
What have you learned?

- Patching my look easy (turn on auto-updates and you’re done)
- But it becomes challenging when done organisation-wide
  - Keeping track of open vulnerabilities
  - Keeping track of already applied patches
  - How to deal with different kind of systems, different management, etc.
  - Patching can affect the stability of systems or service provision
- Develop a patch management policy
Thank you

Any questions?

Next Module: Local Vulnerability Scanning
28\textsuperscript{th} of June 2021
www.geant.org

© GÉANT Association on behalf of the GN4 Phase 2 project (GN4-2).
The research leading to these results has received funding from the
European Union’s Horizon 2020 research and innovation
programme under Grant Agreement No. 731122 (GN4-2).
References:

- Pakiti: https://github.com/CESNET/pakiti-server?
- Patch My PC Home Updater: https://patchmypc.com/home-updater
- RedHat mirror: https://access.redhat.com/solutions/23016
- Debian/Ubuntu mirror: https://www.howtoforge.com/local_debian_ubuntu_mirror
- OpenSuse mirror: https://en.opensuse.org/openSUSE:Mirror_infrastructure