A smorgasbord of tools around Linux at DESY

2FA for IT Linux Admins
Fail2ban
Repository management
Kernel update notifications

Yves Kemp, Joseph Baluyot, Jürgen Hannappel, Jan Engels
Madison, HEPiX Spring 2018, 17.5.2018
- Two-Factor-Authentication (2FA)
- For IT ...
- ... Linux Admins
Why think about 2FA?

- DESY project office required to make use of ISO certified IT services to apply for project management assignments by German ministry
- DESY IT division provides basically all services for project office, as part of the services for all DESY users
- Management and security of administrative logins to IT services is key issue for the ISO certification
- Need to comply with “state-of-the-art” password security
  - In 2016/2017, this meant, amongst others, 2FA
Which systems should be protected by 2FA

Start small:

- Not for ALL DESY users!
- Not for ALL DESY servers!
- Only for (IT) administrators of basic services and their servers
  - Mail, Kerberos, Printing, Config Management, LDAP, Kerberos, Active Directory, …

Start with Linux first

- Windows admin team already has different accounts and computers (VMs) for different tasks: OK for the reviewer (at least for now)
- Linux: `kemp@desktop$ ssh root@important-server` Wide-spread
  - Though usually via puppet-managed .k5login
  - Most urgent need to do something
2FA should have very small impact to daily work

Admin workflows:

• Choice of different second factors possible
• Kerberos/.k5login should remain main source of trust
• #Password entries and #2FA should stay minimal

Server configuration

• Keep changes to secured servers minimal
• Low/no dependency on 2FA service
• Have emergency plans for 2FA system failure

And still make our security department and external auditor happy
Idea: Jump-Hosts

Orthogonal ways for configuration:
- Server sshd-config
- Server local FW config
- Network ACL on switches
- NO 2FA config here!
Which 2FA software? Which 2FA token?

**Software: privacyIDEA**
- Open-source project
- Good community support
- Enterprise edition with support etc. available, we use the community version

**2FA token**
- Hardware token: Yubikey (one-time-passwords)
- Addition: Google Authenticator or other app-based tokens
- SMS and email also available
Server setup

Auth request via https
No WebUI Access!

R-PI in CC Control Room
Isolated (physically, network from/to outside)
2FA Status and Outlook

- Currently: Some systems have configured ssh such that only the jump hosts are allowed
  - Intensive testing
- Unfortunatly, the person implementing the system has left DESY, search for successor is difficult, but once we have someone take up efforts and hold hands with other service providers
- Maybe extend/adapt to Windows
  - Some Web apps: NextCloud e.g.
- Maybe extend/adapt to non-IT administrators
- Learn about 2FA and the involved products in view of a potential broader introduction
- Fail2ban
- for our login nodes
Fail2ban on portal login nodes

- distributed attack detection:
  - collect log files on a log host, analyze there
- central book keeping:
  - database (postgresql) to store active bans
- on logon hosts: client service listens to database,
  - sets ip-tables for bans
- secured communication via database connection
- deployed via puppet
- Repository management
Repository Management: The problem with automated updates

- In theory, automated updates are good
- Most of the time, things work out well
- For special services, exceptions can be configured

- The reality: Automated updates are often disabled, because „one never knows“
- Often no other patch management scheme implemented

- Keep automated updates, but control the repositories
- Basically not a new idea ...
Repository management: Implementation

- Copy a whole repository once per day (hard links)
- Scenario A: Ring
  - Have some soft links that point to (head-n days), and that get renewed every day
  - Configure the repository behind the soft link
  - Emergency break: If a problematic error comes in, stop the renewal until the problem is fixed
  - Fast-forward: If an emergency security patch needs to be deployed at all cost, advance the links
- Scenario B: Manual cluster update through automated updates
  - E.g. for HPC or dCache cluster
  - Configure the repository behind the soft link, e.g. hpc-prod
  - hpc-prod stays for a couple of weeks/month
  - If the cluster needs to be updated: move the soft link (maybe trigger yum update afterwards)
- https://github.com/desyops/repository-timeline
- Kernel update script
Kernel update script

- Assume: Auto-updates are on. Great – but when should I reboot because of new kernel?
  - Our security expert says, I should reboot with every kernel. Well...
- Need to formalize the process of kernel changelog review and evaluation of the impact - also in view of the ISO certification process
- Idea:
  - Have a cronjob that looks at new kernels (SL6, EL7, Ubuntu 14.04, Ubuntu 16.04) in our repositories
  - If there is a new kernel, open a ticket, and enter as information the changelog of the kernel
  - We then need to fill in an evaluation matrix
  - ... And of course react (this is not yet automated, but we are working on parts)
Questions? Comments?