Hardening OpenLDAP on Linux with AppArmor and systemd

- Defense in Depth implemented in Æ-DIR -

Michael Ströder <michael@stroeder.com>

OpenLDAP Developer's Day 2018

Michael Ströder <michael@stroeder.com>

- Freelancer
- Topics the last 20 years
  - Identity & Access Management, LDAP
  - Single Sign-On, Multi-Factor Authentication
  - PKI (X.509, SSH), Applied Crypto
- Open Source / Free Software:
  Æ-DIR, OATH-LDAP, web2ldap

Why?

- We’re all humans and thus error-prone
- Huge software stacks made by humans
- Untrusted input
  (Is there really any trusted input from remote?)
- Prior input validation at lower protocol levels unfeasible

AppArmor (1)

- Linux Security Module (LSM)
- Text-based configuration gets compiled into kernel
- Various CLI tools aa-* also for profile generation
- Profiles, include abstractions and tunables
- Show confinement status: ps -axZ
- Default mode: *targeted*

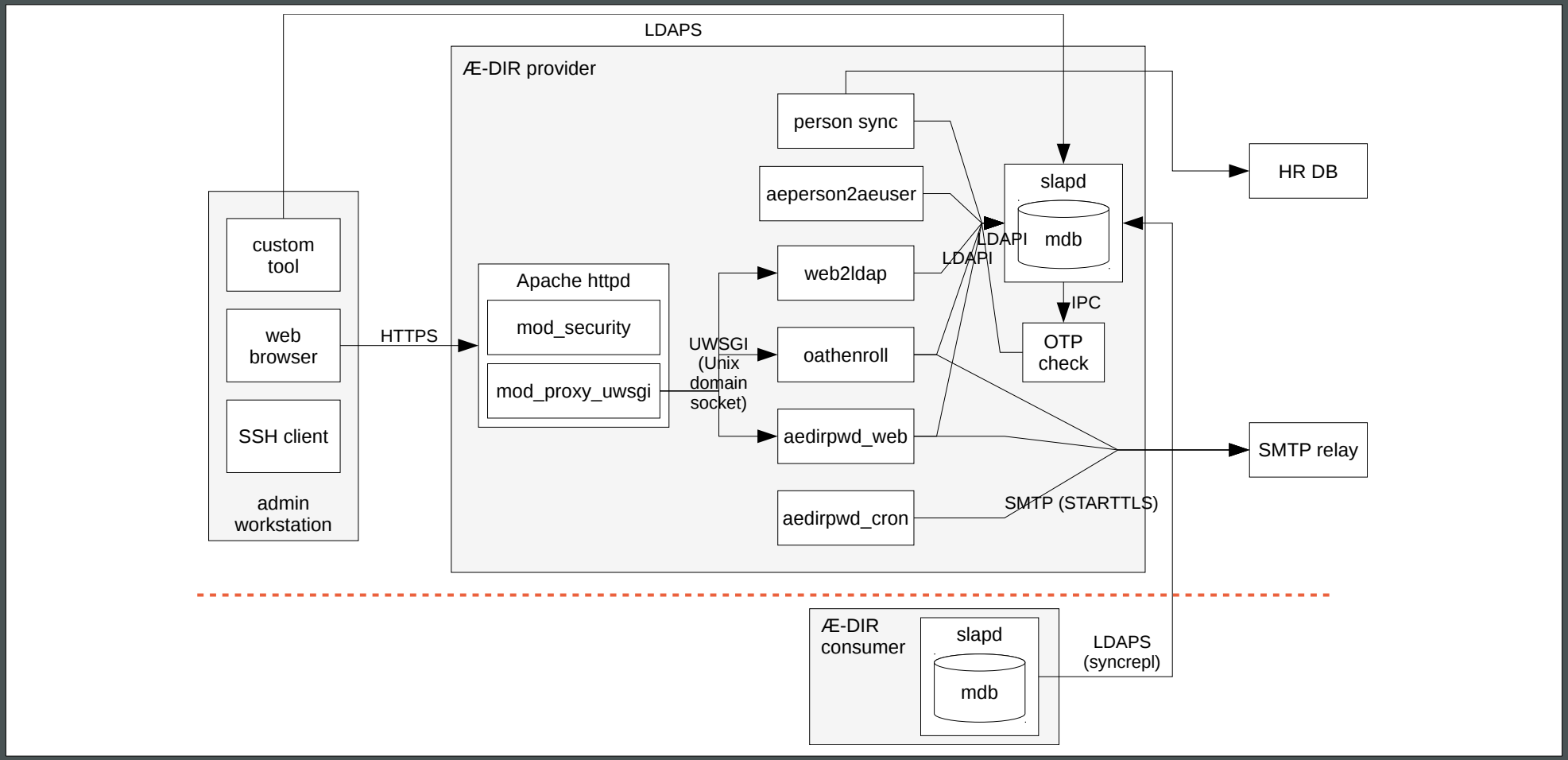
AppArmor (2)

- Restrict permissions:
  - File access (r, w, m, x,)
  - Network access
  - Capabilities(7)
  - rlimit (aka ulimit)
- Not a replacement for input validation, process isolation, file system ownership/permissions, OpenLDAP ACLs, etc.

Environment

- Dedicated machines (bare-metal or VMs)
- Debian Linux / openSUSE Linux / CentOS 7
- All services have separate systemd units
- OpenLDAP 2.4.46, many overlays
- Apache 2.4 as HTTP frontend (TLS termination)
- Some web apps, Python 2.7, separate UWSGI inst.
- Unix Domain Sockets, LDAPI with SASL/EXTERNAL

Æ-DIR Components



Ansible

- Æ-DIR services completely configured with *Ansible*
- All paths already known in Ansible vars
- No OS base configuration!
- Ansible fact for optional AppArmor config:
  ansible_apparmor.status == 'enabled'
- Use Jinja2 templates for profiles and abstractions
- Services and AppArmor must be reliably restarted!

Profile generation

- Automatic profile generation with aa-genprof
- I wanted to understand all this in detail
  → hand-made profiles and abstraction
- 1. Used standard abstractions shipped by OS packages
- 2. Reviewed standard abstractions (yuck!)
- 3. Own base abstractions
- 4. Future: Work with AppArmor maintainer to split and strip standard abstractions

Service ae-slapd

- Profile independent of exec name:
  /etc/systemd/system/ae-slapd.service
  `AppArmorProfile=ae-slapd`
- roles/ae-dir-server/templates/apparmor/ae-slapd.j2
  → /etc/apparmor.d/ae-slapd
- Unused stuff with “deny r” to avoid audit log entries:
  - /etc/ssl/openssl.cnf
  - /etc/gss/mech.d/

Trouble-shooting AppArmor

- auditd is your friend ;-)
  `grep DENIED /var/log/audit/audit.log|ausearch -i`
- Trying to have no false positives when alarming
- systemd units for Python:
  `Environment=PYTHONDONTWRITEBYTECODE=1`
- Sometimes investigating on some strange DENIED lines although everything just works…
- You can easily mask audit message with *deny* but…

systemd (1)

- Much *systemd* concerns, but it's used anyway
- Stumbled on some options in systemd.exec(5):
  - Protect*=
  - Private*= und MountFlags=private
  - NoNewPrivileges=yes
  - SystemCallFilter=…
  - RestrictAddressFamilies=AF_INET AF_INET6 AF_UNIX

systemd (2)

- Uses namespaces(7) and seccomp-bpf
- PrivateDevices=yes connects /dev/log to *journald*
- My /etc/systemd/journald.conf:

```
[Journal]
Storage=none
ForwardToSyslog=yes
```

- YMMV regarding log performance

systemd (3)

- Init scripts and systemd units shipped with OS packages are disabled/removed!
- Separate systemd units allow custom config
- Also helps avoiding incidents caused by over-zealous pseudo config management in OS packages
- Security options configurable by ansible vars

QA

- Enable all protection shields during integration tests
- Disable during pen-Testing in DEV-/QA stage
- ansible role *ae-dir-server*
  - apparmor_enabled: False
  - aedir_systemd_hardening: []
- Re-enable for testing protection against actual findings
- Hardening might help until real security fix is available

Conclusion

- Small effort for me as developer: ~ 3d + ~1.5d
- Separation of components needed
- Still more fine-grained settings left to do
- Hardening does not or only partially protect against: Broken applications, missing updates, etc.
- Configuration management rocks!
- Own customization generally better than OS defaults

:-/

? … !