aehostd

A custom NSS/PAM service for Æ-DIR

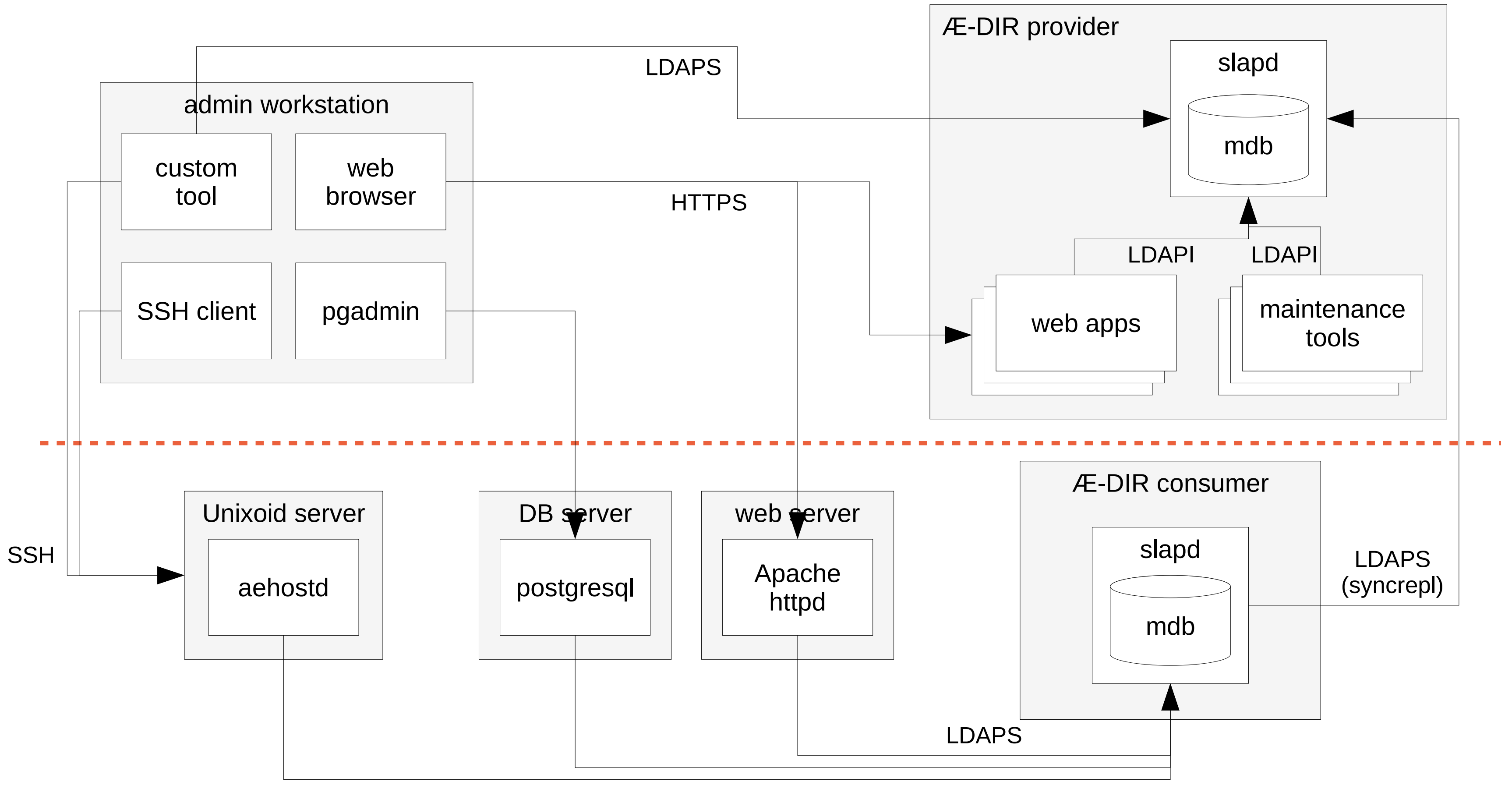
LDAPcon 2019

2019-11-05

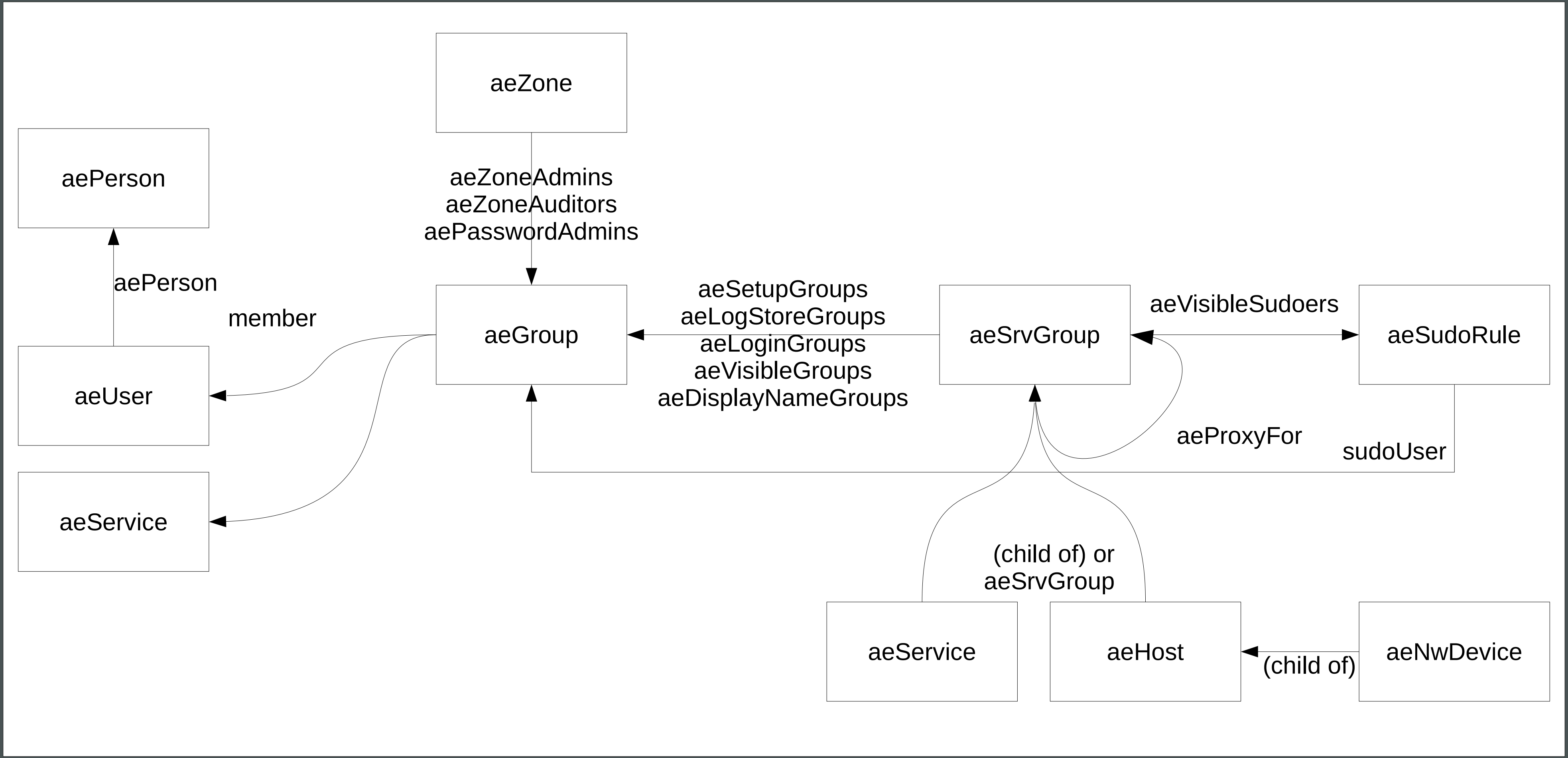
Michael Ströder <michael@stroeder.com>

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

Æ-DIR - 2-tier architecture



Æ-DIR - Entity relationships for access control



Name Service Switch (NSS)

- Config in /etc/nsswitch.conf
- map: module (e.g. passwd: files)
- Modules in shared libs, e.g. /lib/libnss_*.so
- Easy to test with getent map <name>
- Enumeration/caching
- Relevant NSS maps for user management:
  - passwd
  - groups / initgroups

Pluggable Authentication Modules (PAM)

- Config nowadays in /etc/pam.d/
- /etc/pam.d/service refers to shared libs in /lib/security/
- most times common includes are used
- Steps: account, auth, session, passwd
- It's easy to shoot yourself in the foot
- Always keep root shells open during ad-hoc changes
- Always test negative cases! Pen-testing!
- Use config management

sudo

- Privilege escalation
- Configuration:
  /etc/sudoers, usually includes /etc/sudoers.d/*
- Files must have certain ownership permissions
- LDAP schema available (some limits)
- sudo-ldap: separate LDAP connection for each invocation
- sudo via sssd: sudo linked against shared lib of sssd project

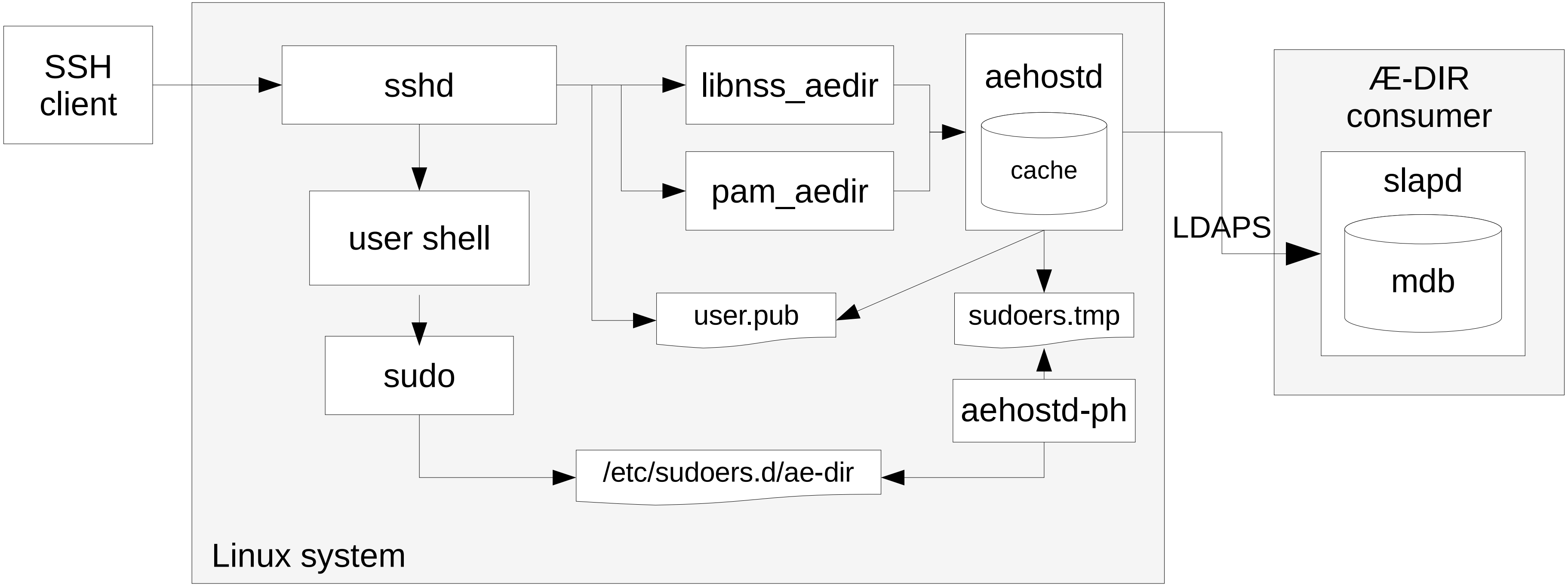
aehostd - Why?

- Æ-DIR’s slapd burns CPU cycles with set-based ACLs
- *sudo-ldap* causing lots of parallel TLS connections
- Connection behaviour
  - unpredictable fail-over order
  - "synced" search operations
- Better automated enrollment needed (host password)
- LDAPI support for NSS/PAM on Æ-DIR servers
- Fed up by asking others for simple features

aehostd - Goals

- Better performance
- Better behaviour for lots of NSS clients:
  - Client-side load-balancing
  - Randomized update timing
- Enrollment automation with pseudo SSH login
- Simplicity:
  Less configuration, less code, less dependencies, less privileges

aehostd / aehost-ph



aehostd - Implementation

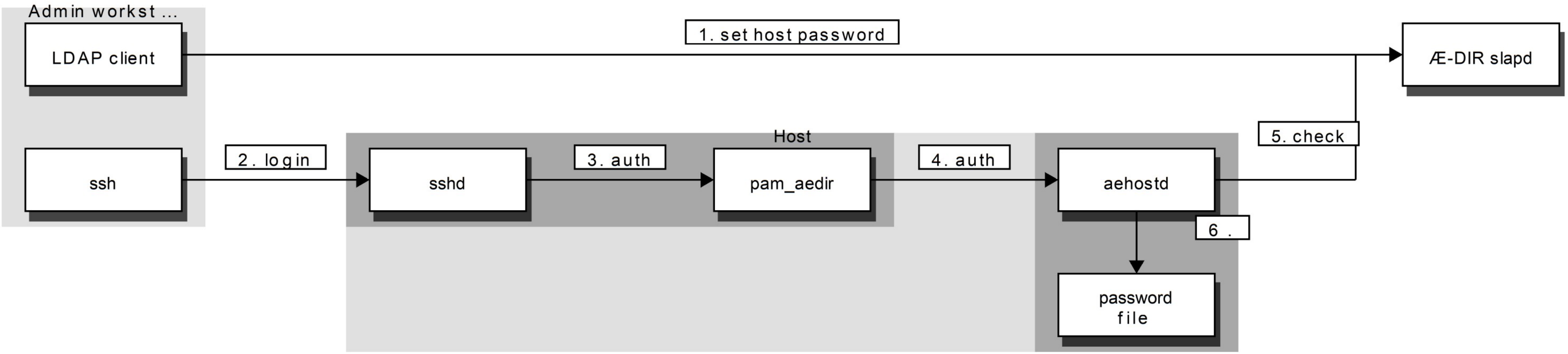
- Unix domain server written in Python
- Uses PAM/NSS front-end modules of *nss-pam-ldapd* preferrably compiled with name "aedir"
- Main service *aehostd* runs as unprivileged user
- Helper service *aehostd-ph* runs as *root* for writing file in */etc/sudoers.d*
- Full map enumeration
- Low-tech sudoers support: Requires CLI tool *cvtsudoers* (sudo 1.8.23+) for converting LDIF to sudoers format

aehostd - Specific Features

- Virtual groups:
  - primary user GIDs
  - role groups
- Syncing of SSH authorized keys
- LDAP session tracking control for better logging
- *hosts* map based on *aeNwDevice* entries
- Enrollment via pseudo login with password

  ```
  ssh aehost-init@host.example.com
  ```

aehostd - Enrollment



aehostd - Configuration

- LDAP URIs, trusted CA cert(s), bind-DN and password
- Separate password file
- *uri_list* vs. *uri_pool*
- Load balancing without external load balancer:
  rotate(uri_pool, hash(FQDN) mod N)
- Example on Æ-DIR servers:
  ```
  uri_list = ldapi://
  uri_pool = ldaps://ae-dir1.example.com ..
  ```
- ansible role available

Performance

- 3000 queries / sec on tiny VM, sufficient for now (max. is 7000 queries / sec with nscd)
- Savings compared to other implementation (extrapolation to 15000 machines, 5 min. refresh):
    - ~ 230 GB / day less LAN traffic
    - ~ 11 GB / day less log traffic (loglevel stats)
- Some more ideas for tuning if really necessary

Conclusion

- Nice results:
  - Decent performance even with Python
  - Less resource usage
  - Seems to be quite stable
- PAM is scary...
- Freedom to implement features
- But have to avoid featuritis!
- To-do: Python 3 (end of 2019), salt state, puppet module

:-/

? … !