HOMEWORK 4

LDAP

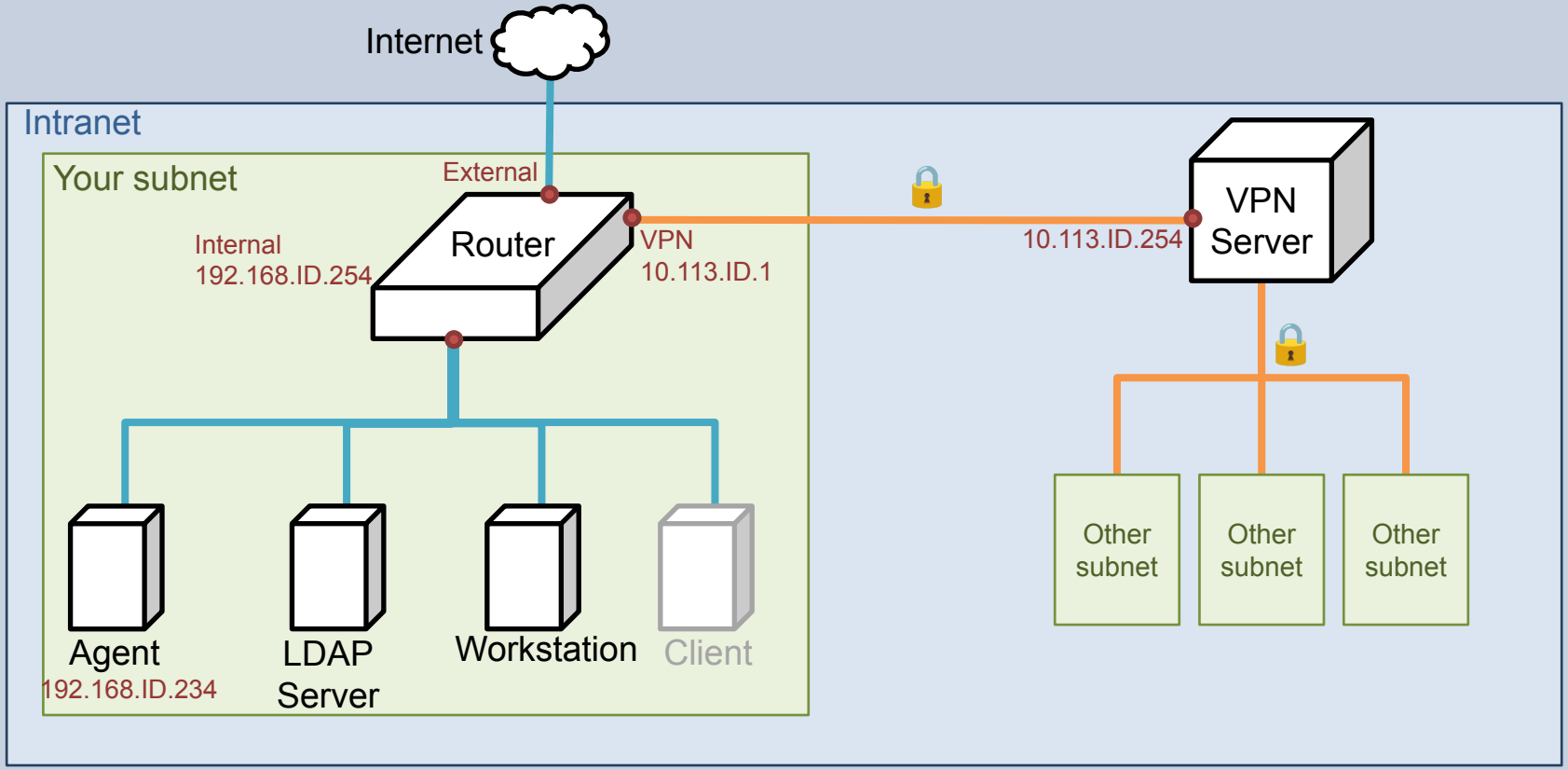
wkhsiao

國立陽明交通大學資工系資訊中心
Information Technology Center, Department of Computer Science, NYCU

Objectives

- Build a basic LDAP service
- Understand how to…
  - configure LDAP server
  - manage LDAP data using LDIF
  - auth and permission control on Unix client with LDAP server
  - customize your own objectClass and using OLC(on-line configuration)

Overview - Architecture

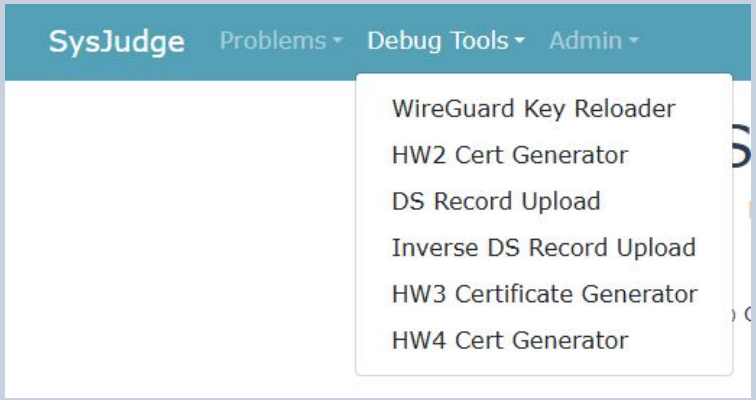


Overview (cont.)

- A simple LDAP server
  - LDAP client
- One or more Workstations
  - LDAP client

Requirements

- LDAP Server
  - IP: 192.168.ID.y/24 with static DHCP, where y is arbitrary.
  - Hostname: ldap.{ID}.nasa. (4%)
  - Base DN: dc=<ID>, dc=nasa
  - LDAPS and force TLS search (8%)
    - Not LDAP over TLS (StartTLS) (2%)
    - Use HW4 certificate generator to get your key and certificate



Requirements

- Workstation
  - IP: 192.168.ID.y/24 with static DHCP, where y is arbitrary
  - Hostname: workstation.{ID}.nasa. (4%)

Requirements

- Organizational Unit Naming
  - People
  - Group (posixGroup)
  - Ppolicy
  - SUDOers
  - Fortune (our customize objectClass)

Requirements

We need two posix group in LDAP:

- ta group (GID=10000)
  - can login (ssh) into LDAP server and any workstations (6%)
  - can use sudo for any command (7%)
    - ex. `sudo adduser`
- stu group (GID=20000)
  - can login (ssh) into workstations, cannot login into LDAP server (6%)
  - only allow sudo for `ls` command (7%)
- You need use “LDAP” to implement above requirements
  - Including sudo rules and ssh key!
- TA will add any named user using generalta into these group (10%)

Requirements

Add an user with DN “uid=generalta,ou=People,<Base DN>”

- This user under ta group, use ta group permission
- Allow this user to connect via SSH with both ssh public key and password
  - uid: generalta
  - uid number: 10000
  - public key: <ta’s public key> # See p.10
  - user password: <your TA_PASSWORD> # Same as HW3
    - user password need hash

Requirements

TA's public key: https://nasa.cs.nycu.edu.tw/na/2025/slides/hw4.pub

- Public key:

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIFfg2DMY3DfBBvZCnqN8Az5tUnVQca+qXkJ9HceOcRAy 2025-na-hw4
```

- User can set their authorized keys with the sshPublicKey attribute

Requirements

Add another user with DN “uid=stu<ID>,ou=People,<Base DN>”

- This user under stu group, use stu group permission
- Allow this user to connect via SSH with both ssh key and password
  - uid: stu<ID>
    - e.g. stu1, stu55
  - uid number: 20000 + <ID>
    - e.g. 20001, 20055
  - user password: <your TA_PASSWORD>

Requirements

- Configure LDAP Client on every machine
  - Configure LDAP for login (ssh) authentication
    - can use password or public key to login
  - When you add a user into LDAP, this user can login on any workstation or LDAP Server
    - Login permissions at Page 8

Requirements

- Set proper LDAP access control
  - Allow generalta to manage users and groups
  - Allow every users to modify their own userPassword, loginShell and sshPublicKey (6%)
    - Set other attributes as read-only (6%)
  - Allow users to search all user data but other users' password (6%)
    - i.e., users can only read their own password
    - generalta can write to it but not read!

Requirements

- Set password policy for each user (10%)
  - userPassword can’t be same as previous when change password
    - But can set password as previous two time used
    - You need implement this by LDAP way
  - password requires at least 8 characters long
  - password must contains at least 3 different classes of characters:
    - Upper-case characters
    - Lower-case characters
    - Digits
    - Special characters
  - Hint: ppolicy overlay & pwdCheckModule

Requirements

- Add an OU(Fortune) that contains fortunes (4%)
- Add an ObjectClass fortune with **on-line configuration (OLC)** (6%)
  - **schema**
    - objectClass's oid should be under the UUID branch
    - extend from **top**, add **author** field(octetString), and **id** field(integer)
    - **author's** matching, substring and order should be "**case insensitive, space insensitive**"
    - use existing **description**(RFC 4519) attribute to place sentences
  - we would check whether this objectclass is in database (cn=config)

Requirements

- Import fortunes (4%)
  - from given yaml file (link)
  - 3 fields
    - ID
    - Author
    - Description
- enable features (4%)
  - server side sorting
  - pagination
  - Hint: slapo-sssvlv

```
-
  ID: 106
  Author: Richard Feynman
  Description: 'I don''t know what''s the matter with people: they don''t learn by understandin
-
```

```
dn: cn=fortune-1,ou=Fortune,dc=254,dc=nasa
objectClass: fortune
objectClass: top
cn: fortune-1
author: Richard Feynman
id: 1
description: The first principle is that
you must not fool yourself -- and you
 are the easiest person to fool.
```

Requirements

- Configure the HW3 mail server to use LDAP for authentication.
  - Ensure that LDAP users under ou=People,dc=<ID>, dc=nasa can:
    - Send emails (bonus 5%)
    - Receive emails works) (bonus 5%)
    - This requires that SMTP, IMAP, and POP authentication succeed, and that mail delivery functions properly.

Attention

- Your work will be scored by Online Judge system
  - Online Judge cools down for **several minutes** after judge
  - Only the **LAST** submission will be scored
  - Late submission will **NOT** be accepted
- **ALWAYS BACKUP** your system before submission, as we may do malicious actions
- Make sure everything works after reboot
- **Deadline:** **5/29 (Thu) 23:59**