國立陽明交通大學資訊工程學系資訊中心
IT Center of Department of Computer Science, National Yang Ming Chiao Tung University

Web Service

Computer System Administration Homework 3
Bogay, NYCU CSIT

Last updated: 2025/11/18 13:38

Outline

- Public Key Infrastructure
- Web Server
- OpenID Connect
- Matrix Homeserver
- Web Service Development
- Database
- OpenTelemetry

Setup - User (5%)

- Based on previous homeworks
- Make your **judge** user a **no password sudoer**
  - i.e. judge should be able to sudo any command without a password prompt
- Hint:
  - visudo(8)
  - sudoers(5)
  - NOPASSWD

Setup - Virtual Hosts (3%)

- There are some hosts on your server (`ID` is your student ID, e.g. `312551016`):
  - `{ID}.cs.nycu`
  - `hello.{ID}.cs.nycu`
  - `acme.{ID}.cs.nycu`
  - `auth.{ID}.cs.nycu`
  - `matrix.{ID}.cs.nycu`
  - `mas.{ID}.cs.nycu`
  - `i.{ID}.cs.nycu`
  - `postgres.{ID}.cs.nycu`
- TA will use `getent` command to test this, so please ensure it is available to the `judge`.

Hint: `/etc/hosts`

Setup - Docker Compose (4%)

- You must use Docker Compose to manage all services for this homework.
  - Ensure Docker and Docker Compose are installed on your system.
- **Compose File (1%)**: Place your configuration at `` `/home/judge/hw3/deploy/docker-compose.yml` ``. It must define the following services:
  - `` `web` ``: A reverse proxy for all services except `` `postgres` ``.
  - `` `acme` ``: The ACME server.
  - `` `auth` ``: The OpenID Provider (OP).
  - `` `matrix` ``: The Matrix Synapse homeserver.
  - `` `mas` ``: The Matrix authentication service.
  - `` `uurl` ``: The URL shortener.
  - `` `postgres` ``: The PostgreSQL database.
- **Automatic Restart (1%)**: Configure your services to restart automatically if they crash or after a Docker service reboot.
- **Service Health (1%)**: Your services must be fully functional after a `` `docker compose down` `` followed by a `` `docker compose up -d` ``.
- **Networking (1%)**: You are only allowed to use the default bridge network created by Docker Compose. Do not create other custom bridge networks.

Web Server (8%)

- Make `hello` respond raw text: `I love NYCU NASA 2025`. (1%)
- The responses should not contain `Server`, `X-Powered-By` and `Via` headers. (2%)
- Redirect HTTP to HTTPS for all hosts. (2%)
- Enable HTTP Strict Transport Security (HSTS) for all hosts. (2%)
- Enable HTTP/3 support for all hosts. (1%)

PKI (7%)

- Create your own CA as root CA for your server, and: (3%)
  - Put key at `` `/home/judge/hw3/sarootca.key` ``.
  - Put cert at `` `/home/judge/hw3/sarootca.crt` ``.
  - Set subject and issuer to `` `C=TW, O=National Yang Ming Chiao Tung University, CN=SA{ID} Root CA` ``. (The order matters)
- Sign an intermediate CA from your root CA, and: (3%)
  - Put key at `` `/home/judge/hw3/sa.key` ``.
  - Put cert at `` `/home/judge/hw3/sa.crt` ``.
  - Set subject to `` `C=TW, O=National Yang Ming Chiao Tung University, CN=SA{ID}` ``.
  - Restrict the certificate chain length so that certificates derived from it cannot be used as CA.
- Trust root CA on your system. (1%)

ACME Server (10%)

You will host your own ACME server to automate certificate issuance for your services. We recommend using an existing implementation like step-ca for this task. But any implementation that meet the requirements is acceptable.

- **Hosting and Challenge Type (4%)**:
  - Host the ACME server at `` `acme.{ID}.cs.nycu` ``.
  - The server must be configured to use your intermediate CA (`` `SA{ID}` ``) to sign certificates.
  - It only needs to support the http-01 challenge type.
  - Store ACME server URL in `` `SA_ACME_SERVER_URL` `` environment variable. TA will try to use certbot to get a certificate.
  - You should ensure your server can resolve `` `ta.{ID}.cs.nycu` `` to `` `192.168.255.123` ``.
- **Issuance Policy (3%)**:
  - The server must only issue certificates for `` `{ID}.cs.nycu` `` and its subdomains. Requests for any other domains must be rejected.
  - TA only test this using `` `traefik.me` `` domain, you should at least you don't issue certificates for this domain.
- **Certificate Constraints (2%)**:
  - Certificates must have a short validity period, not exceeding 30 minutes.
- Certificates must be issued for a single domain only (1%)

Note: The judge uses `` `ssh <cmd>` `` to execute `` `<cmd>` `` on your server.
Please ensure the environment you set is accessible in this situation.

OpenID Provider (10%)

- You will host an OpenID Connect (OIDC) Provider to handle authentication. I recommend using a certified OIDC implementation like Keycloak or Hydra, but any compliant provider is acceptable.
- You should read the complete homework spec and decide which one to use.
- I use kanidm, BTW, it was a struggle.

OpenID Provider (10%)

- Host the OIDC Provider at `https://auth.{ID}.cs.nycu`.
- Create a confidential client for machine-to-machine communication:
  - `SA_OIDC_PROVIDER`: The full URL where the OIDC discovery document is exposed.
  - `SA_OIDC_CLIENT_ID`
  - `SA_OIDC_CLIENT_SECRET`
- The provider must expose a valid OIDC discovery document that meets the following requirements:
  - The `issuer` field matches `https://auth.{ID}.cs.nycu`, URL with path is allowed.
  - The `token_endpoint`, `userinfo_endpoint`, `introspection_endpoint` and `revocation_endpoint` are present and use HTTPS.
- TA will use those endpoints to create, revoke and inspect access token.

Matrix Homeserver Setup (2%)

- Host a Synapse homeserver at `matrix.{ID}.cs.nycu`.
- The server's logical server name must be set to `{ID}.cs.nycu`.
  - This will be checked in later test cases
- The Client-Server API must be available. A GET request to `https://matrix.{ID}.cs.nycu/_matrix/client/versions` should return a successful response.

Matrix Authentication Service (MAS) (3%)

- Host a MAS at `mas.{ID}.cs.nycu`.
- Provide `SA_MAS_CLIENT_ID` and `SA_MAS_CLIENT_SECRET` for judge to interact with MAS admin API.
- TA will check server name in this testcase.

SSO with OpenID Connect (5%)

- Register the MAS as a confidential client in your OpenID Provider.
- Configure Synapse and MAS to allow users to log in via your OIDC provider.
- A user named `judge` must connected to your OpenID Provider.
- TA will check this by MAS admin API.

URL Shortener (2%)

- Host a URL shortener at `` `i.{ID}.cs.nycu` ``
- There are only three routes:
  - `` `POST /api/shorten` ``: create a short URL
  - `` `GET /-/{code}` ``: redirect to original URL
  - `` `GET /metrics` ``: report metrics in prometheus format
- You can implement the API in any tech stack.
- You should use PG to store those URL mappings with following schema in database named `` `uurl` ``
- Implement `` `create_short_url` `` and `` `redirect` `` endpoints. (2%)
  - See OpenAPI spec for more details

```
CREATE TABLE urls (
    id SERIAL PRIMARY KEY,
    creator VARCHAR(255),
    original_url TEXT NOT
NULL,
    short_code VARCHAR(10)
UNIQUE NOT NULL,
    created_at TIMESTAMP WITH
TIME ZONE DEFAULT
CURRENT_TIMESTAMP
);
```

Database Schema

URL Shortener (3%)

- The URL shortener should provide metrics in prometheus format.
- There are three metrics you should export, all of them are counters (2%)
  - `` `sa_redirects_total` ``: number of successful redirects
  - `` `sa_redirects_not_found_total` ``: number of redirects failed with user given wrong short code
  - `` `sa_created_urls_total` ``: number of short URLs created by your service
- The metrics should have `` `container` `` label, the value is the first 12 chars of the container ID (1%)

URL Shortener (6%)

- The `/api/shorten` and `/-/{code}` should be protected by OIDC. Only requests with a valid bearer token issued by your OP in `Authorization` header can access those routes. (4%)
    - TA will use client credential flow to get an access token with credential described in previous OpenID Provider section.
- The `/metrics` endpoint should be protected by basic auth. (RFC 2617) (2%)
    - Username: `sa`
    - Password: `{ID}`

Hint: oauth2-proxy

Matrix Bot (5%)

- Create a Matrix bot account named `@uurl:{ID}.cs.nycu` (1%)
- Ensure your bot can autojoin rooms after being invited (1%)
- Let the bot listen to messages with following pattern:
    - `!sa shorten <url>` (1%): if `<url>` is a valid URL, reply the short URL directly, otherwise reply `Usage: !sa shorten <url>`.
    - `!sa get <short_code>` (1%): if we can find original URL by given `<short_code>`, reply it directly, otherwise reply `Usage: !sa get <short_code>`.
    - `!sa list` (1%): list all URLs created by this matrix user. Each line should follow format: `<short_code> => <url>`, e.g. `X0h4VR => https://site.nasa.cs.nycu.edu.tw`
    - for other strings starts with `!sa`, reply `Usage: !sa {shorten|get|list}`, otherwise ignore them (1%)

Database (10%)

- Host a PostgreSQL 17 server at `postgres.{ID}.cs.nycu:5432` (2%)
- Create a user `judge` with password `judge` who has all permission to the database `uurl`. (1%)
- Make your PostgreSQL server support TLS connection. (3%)
- Make user `judge` can only authenticate via client cert with `CN=judge` when login from remote. (4%)

Access Log - Basic (6%)

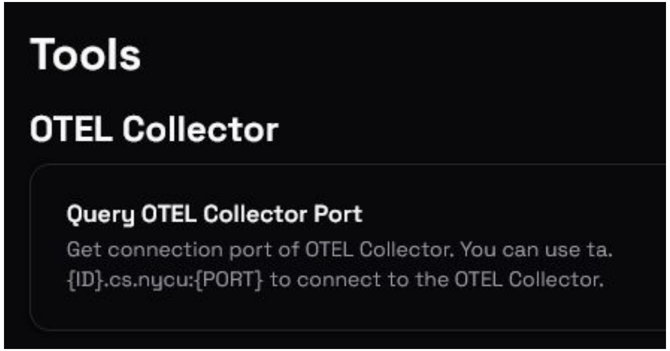
- Write access logs to `/home/judge/hw3/logs/access.log`.
- No strict format, but should contain: (2%)
  - URI, e.g. `/.well-known/openid-configuration`
  - User agent, e.g. `curl/8.5.0`
  - Host, e.g. `hello.312551016.cs.nycu`
- Should not log when request user agent is `sa-probe/{ID}`. (4%)

OpenTelemetry Logging (8%)

Send access log to TA's OTel collector in OTLP http format. The receiver serves TLS signed by your CA. You can use Tools on online judge to get connection information of your account.

- Each log must contain following attributes (8%): `sa.uri`, `sa.host`, `sa.method`, `sa.user_agent`, `sa.status`. Logs should not be sent if user agent is `sa-probe/{ID}`.
- You must authenticate via client certificate to send log (2%)

Hint: vector, fluentbit, OTel Collector



Tools

OTEL Collector

Query OTEL Collector Port

Get connection port of OTEL Collector. You can use ta.{ID}.cs.nycu:{PORT} to connect to the OTEL Collector.

Remind

- Check which IP your service listen on
- Check certificates trusted by your container / application
- Check file system permission
- Don't trust LLM output without consulting official doc
- Sometimes reading source code is the fastest way to check the application behavior
- Use automation tools to help you operate services
  - e.g. Ansible

Attention

- Deadline: 12/23 (Tue.) 23:59
- Your work will be scored by Online Judge system
  - Only the **LAST submission** will be **scored**
  - Late submission will **NOT** be accepted
- We will try to fetch your service config and machine info for auditing

Attention

- **ALWAYS BACKUP** your system before submission
  - We may do malicious actions (e.g. Drop your DB, Delete containers)
- TAs reserve the right of final explanations.
  - Specs and the points of each subjudges are subject to change in any time. (with notification)
- Make sure everything works after reboot

Note

- TA hours will be available for this homework. Time will be announced to Google Groups and E3.
- Ask if you think the spec or judge output is ambiguous.
    - You can ask us to provide more information in judge script output; we might consider it.

Help

- Join NCTUNASA google group
  - If you have any question, you can post your problem in this group, TAs and Students will help you.
  - https://groups.google.com/g/nctunasa
- UNIX 常見指令教學
  - https://it.cs.nycu.edu.tw/unix-basic-commands
- How To Ask Questions The Smart Way
  - https://github.com/ryanhanwu/How-To-Ask-Questions-The-Smart-Way