NA 2024 Final Project

Lin Lee

國立陽明交通大學資工系資訊中心

Computer Center of Department of Computer Science, NYCU

Introduction and Purpose

- This homework has 15% credit.
  - 5% for basic review: Router, DNS, Mail
  - 10% for advanced system configurations: ELK, monitoring system.
- The goal is to build an intranet that provides several services, including NAT, VPN, DNS, Mail, WWW, etc.
- Know what you should know about configuring and managing these services.

Overview (1)

- Create an intranet which contains several VMs:
  - “Router” (1 Public IP and 1 Private IP)
    - Provides NAT for the VMs under your LAN without public IP and VPN.
    - Can connect to all VMs inside your LAN.
    - Can connect to all VMs inside your LAN via VPN.
  - “DNS Server” (1 Public IP and 1 Private IP)
    - Authoritative name server for your own domain.
    - We expect you to use nycu.me service, but if you have your own domain, it is ok :)

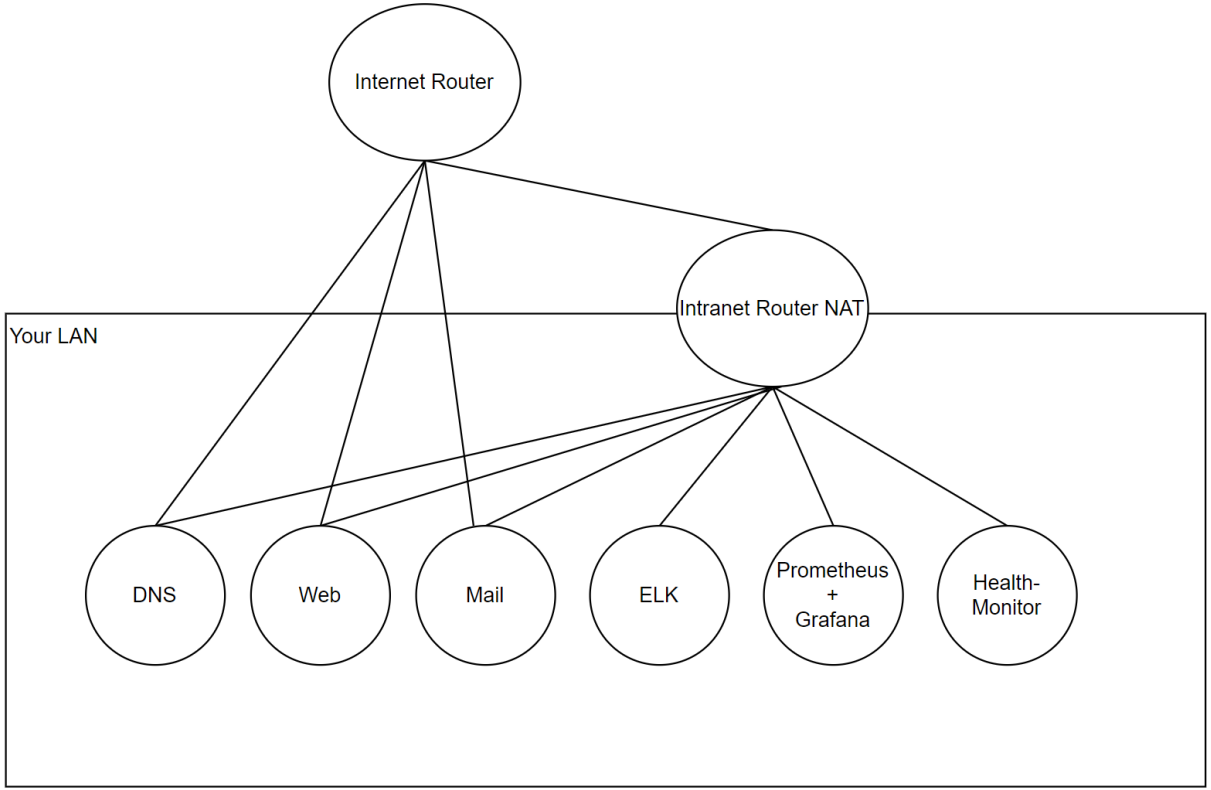
Overview (2)

- Create an intranet which contains several VMs:
  - “Web” (1 Public IP and 1 Private IP)
    - Provides web service.
    - You can host an arbitrary website like the default page from your web server.
    - Or you can host your own personal website if you want :)
  - “Mail” (1 Public IP and 1 Private IP)
    - An email server which could send and receive mail.
    - We will use this mail server for monitor’s alerting.

Overview (3)

- Create an intranet which contains several VMs:
  - “ELK” (1 Private IP)
    - Receives logs from DNS server and web server.
    - Processes the logs.
    - Visualizes the logs and provides a dashboard for admin.
  - “Prometheus + Grafana” (1 Private IP)
    - Collects metrics information from DNS server and web server.
    - Visualizes the metrics information and provides a dashboard for admin.
  - “Health-Monitor” (1 Private IP)
    - Monitors service’s health status, sending email when service is down.

Topology


Internet Router
Intranet Router NAT
Your LAN
DNS
Web
Mail
ELK
Prometheus
+
Grafana
Health-
Monitor

Requirements – Domain Name

- "Domain Name"
  - **You have to have a domain name to continue this homework**.
  - You can register one free third-level domain name on https://nycu.me by NYCU ID.
  - Or you can use your own domain name, that's okay.
  - In the remaining slides, we assume your domain name is "{domain}.nycu.me".
  - If you are not a NYCU student, and you have no domain name, please contact TAs.
    - Choose a domain name you like (length >= 4), and tell TAs.
    - List the records you need to add, and tell TAs.
    - Generate the DNSKEY records, and tell TAs.

Requirements – Router

- “Router”
  - **The hostname will be set to “router”.**
  - This VM will have these network interfaces:
    - External: Internet facing
      - Provide NAT on this interface. Packets from your subnet can go to Internet through this interface.
      - IP: Given.
    - Internal: To your LAN.
      - IP: Given (10.1.2.254), but you can change it if you want :)
  - The sshd service will be enabled for you to connect to your LAN
  - Your VMs in your LAN without public IP should be able to access internet via NAT. (0.5pt)
  - You have to build up a VPN service (0.5pt).
    - After building up the VPN service, you can directly connect to your LAN via VPN, without ssh into router as a jump host.

Requirements – DNS (1)

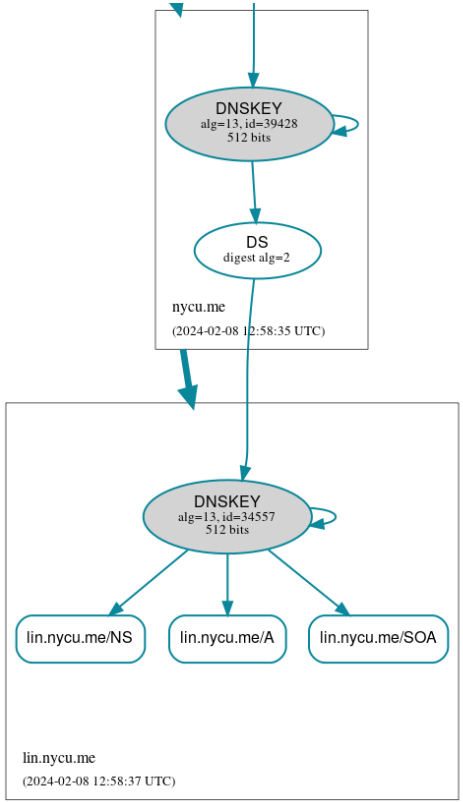
- “DNS”
  - **The hostname will be set to “dns”.**
  - This VM will have these network interfaces:
    - External: Internet facing
      - Provides DNS resolving service.
      - IP: Given.
    - Internal: To your LAN.
      - IP: Given (10.1.2.10), but you can change it if you want :)
  - The sshd service will be enabled.

Requirements – DNS (2)

- Basic DNS resolving (0.25pt):
  - Zone: {domain}.nycu.me (Or your own domain)
    - nameserver: ns.{domain}.nycu.me
    - ns.{domain}.nycu.me A {DNS_given_IP}
    - {domain}.nycu.me A {Web_given_IP}
    - www.{domain}.nycu.me A {Web_given_IP}
- Basic security configuration (0.75pt):
  - As an Authoritative-Only DNS server, set the right setting for the recursion queries.
  - To prevent unexcepted RR replcation, allow nobody except localhost to send axfr.
- DNSSEC (1pt):
  - You can use any method to finish this part.
  - We will use this tool to check your DNSSEC:
    - https://dnssec-debugger.verisignlabs.com/

DEMO - Correct DNSSEC Result

.	Found 2 DNSKEY records for . DS=20326/SHA-256 verifies DNSKEY=20326/SEP Found 1 RRSIGs over DNSKEY RRset RRSIG=20326 and DNSKEY=20326/SEP verifies the DNSKEY RRset
me	Found 1 DS records for me in the . zone DS=45352/SHA-256 has algorithm RSASHA256 Found 1 RRSIGs over DS RRset RRSIG=30903 and DNSKEY=30903 verifies the DS RRset Found 3 DNSKEY records for me DS=45352/SHA-256 verifies DNSKEY=45352/SEP Found 1 RRSIGs over DNSKEY RRset RRSIG=45352 and DNSKEY=45352/SEP verifies the DNSKEY RRset
nycu.me	Found 1 DS records for nycu.me in the me zone DS=39428/SHA-256 has algorithm ECDSAP256SHA256 Found 1 RRSIGs over DS RRset RRSIG=12693 and DNSKEY=12693 verifies the DS RRset Found 1 DNSKEY records for nycu.me DS=39428/SHA-256 verifies DNSKEY=39428/SEP Found 1 RRSIGs over DNSKEY RRset RRSIG=39428 and DNSKEY=39428/SEP verifies the DNSKEY RRset
lin.nycu.me	Found 1 DS records for lin.nycu.me in the nycu.me zone DS=34557/SHA-256 has algorithm ECDSAP256SHA256 Found 1 RRSIGs over DS RRset RRSIG=39428 and DNSKEY=39428/SEP verifies the DS RRset Found 1 DNSKEY records for lin.nycu.me DS=34557/SHA-256 verifies DNSKEY=34557/SEP Found 1 RRSIGs over DNSKEY RRset RRSIG=34557 and DNSKEY=34557/SEP verifies the DNSKEY RRset ns1.lin.nycu.me is authoritative for lin.nycu.me lin.nycu.me A RR has value 103.179.29.42 Found 1 RRSIGs over A RRset RRSIG=34557 and DNSKEY=34557/SEP verifies the A RRset



Requirements – Mail (1)

- “Mail”
  - **The hostname will be set to “mail”.**
  - This VM will have these network interfaces:
    - External: Internet facing
      - Provides email service.
      - IP: Given.
    - Internal: To your LAN.
      - IP: Given (10.1.2.20).
  - The sshd service will be enabled.
  - Mail domain:
    - {domain}.nycu.me
    - mail.{domain}.nycu.me
  - Basic mailing functionality. (0.5pt)

Requirements – Mail (2)

- Basic DNS resolving:
  - Zone: {domain}.nycu.me (Or your own domain)
    - mail.{domain}.nycu.me A {Mail_given_ip}
    - Set MX record.
      - Sending email to {domain}.nycu.me will go to mail.{domain}.nycu.me
    - SPF
      - Allow only your server to send mails using your domain
      - Deny other servers from pretending you, and drop these invalid mail
- Basic security configuration (0.5pt):
  - STARTTLS on IMAP/SMTP: Use self-signed certificate (Or use Let's Encrypt)
  - User Authentication on IMAP/SMTP:
    - Only send emails with authenticated username@
    - Avoid to fake other users on envelop from
  - No Open Relay

Requirements – Mail (3)

- Outgoing Mail :
  - DKIM (1pt)
    - Signing your outgoing email with your private key
    - A DNS TXT record for DKIM
    - <selector>._domainkey.{domain}.nycu.me. IN TXT <DKIM-Information>
  - DMARC (0pt)
    - Let others drop mails that does not pass DMARC policy check
    - _dmarc .{domain}.nycu.me. IN TXT <DMARC-Rules>
- Incoming Mail (0pt):
  - Do SPF policy check on incoming email
  - Do DKIM policy check on the incoming email
  - Do DMARC policy check to the incoming email

Requirements – Web

- “Web”
  - **The hostname will be set to “mail”.**
  - This VM will have these network interfaces:
    - External: Internet facing
      - Provides email service.
      - IP: Given.
    - Internal: To your LAN.
      - IP: Given (10.1.2.30).
  - The sshd service will be enabled.
  - You can host an arbitrary website like the default page from your web server.

Requirements – ELK (1)

- “ELK”
  - **The hostname will be set to “elk”.**
  - This VM will have these network interfaces:
    - Internal: To your LAN.
      - IP: Given (10.1.2.40).
  - The sshd service will be enabled.

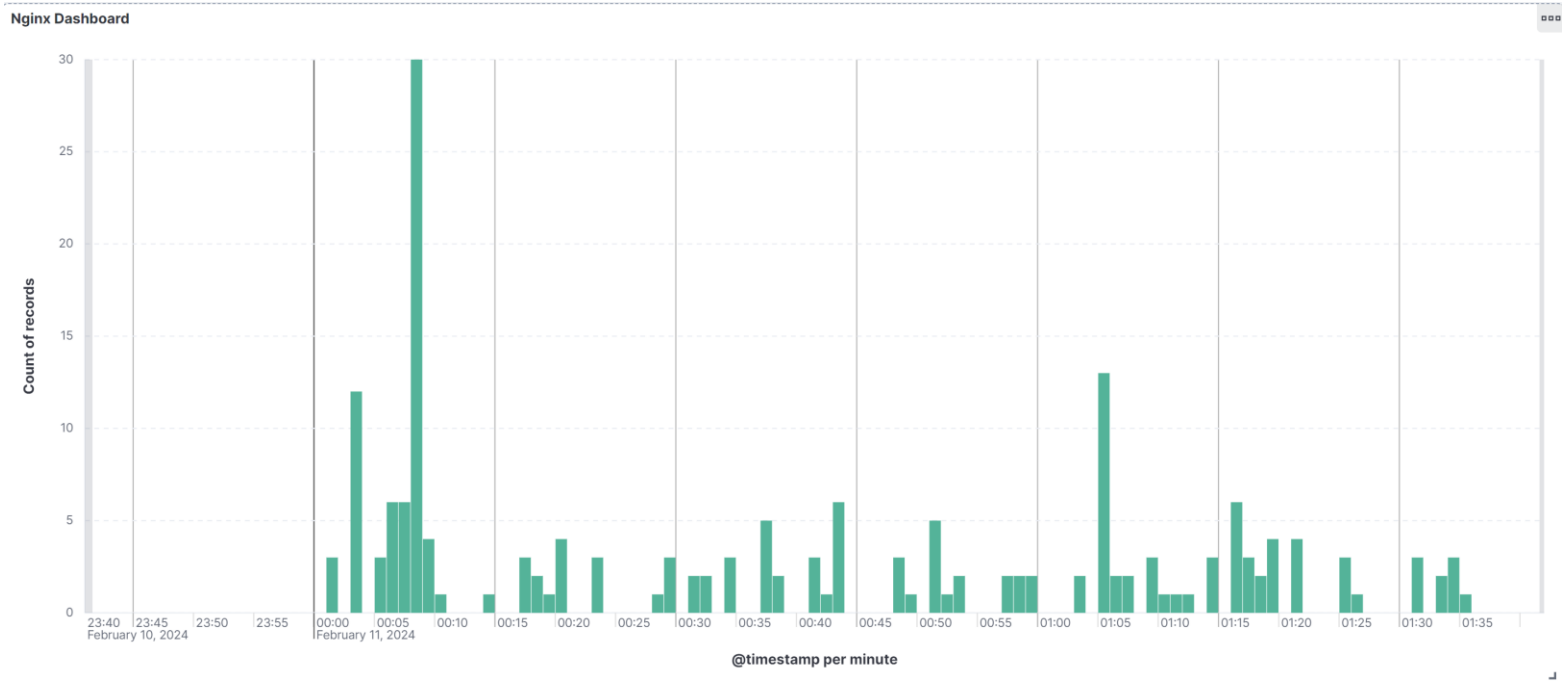
Requirements – ELK (2)

- Logstash:
  - Collecting logs from web server or DNS server.
    - Only collecting accessing(querying) logs from the server. (0.5pt)
    - Also collecting sudo logs from the server(from one is enough). (0.5pt)
- Basic configuration:
  - Log index from server A with service B on date C named “A-B-C”, e.g. “dns-sudo-2024.02.10”.
  - Parse the log with filter and grok if necessary :)

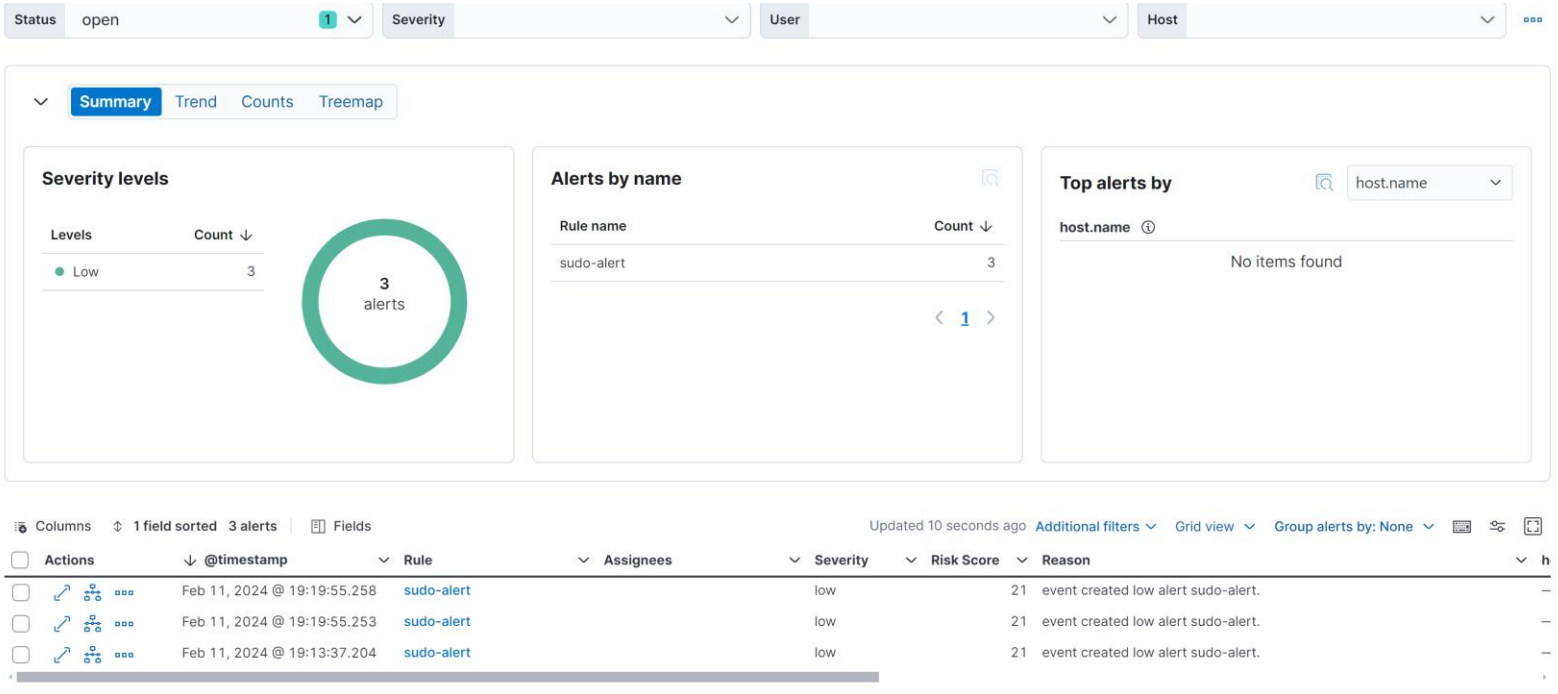
Requirements – ELK (3)

- Elasticsearch:
  - Receives logs from Logstash. (1pt)
- Kibana:
  - Add a panel for your service showing the number of accessing or querying. (1pt)
  - Add an alert for somebody using sudo (using SIEM). (1pt)
    - Write into logs when an alert is raised.
    - Sending mail is better if you are able to buy a license, but, don't waste money :)
  - You can protect your VM by installing agent on it and integrated into ELK. (0pt)
  - Enhancing Malware Detection: Endpoint Detection and Response Solutions with Elastic SIEM
- You can use arbitrary solutions to do such thing, but we require you:
  - Correctly send logs from your VMs to this server.
  - Correctly visualize the logs.

Demo – ELK (1)


Nginx Dashboard
Count of records
30
25
20
15
10
5
0
23:40
February 10, 2024
23:45
23:50
23:55
00:00
February 11, 2024
00:05
00:10
00:15
00:20
00:25
00:30
00:35
00:40
00:45
00:50
00:55
01:00
01:05
01:10
01:15
01:20
01:25
01:30
01:35
@timestamp per minute

Demo – ELK (2)


Status
open
1
Severity
User
Host
Summary
Trend
Counts
Treemap
Severity levels
Levels
Count
Low
3
3
alerts
Alerts by name
Rule name
Count
sudo-alert
3
1
Top alerts by
host.name
host.name
No items found
Columns
1 field sorted
3 alerts
Fields
Updated 10 seconds ago
Additional filters
Grid view
Group alerts by: None
Actions
@timestamp
Rule
Assignees
Severity
Risk Score
Reason
Feb 11, 2024 @ 19:19:55.258
sudo-alert
low
21
event created low alert sudo-alert.
Feb 11, 2024 @ 19:19:55.253
sudo-alert
low
21
event created low alert sudo-alert.
Feb 11, 2024 @ 19:13:37.204
sudo-alert
low
21
event created low alert sudo-alert.

Requirements – Prometheus + Grafana

- “Prometheus”
  - **The hostname will be set to “prometheus”.**
  - This VM will have these network interfaces:
    - Internal: To your LAN.
      - IP: Given (10.1.2.50).
  - The sshd service will be enabled.
  - Please build up Prometheus and Grafana service.
  - Collects metrics information from web server or DNS server (from one is ok).
  - Provides a dashboard for CPU usages, memory usages, and the network traffic statics for your server. (2 pt)
  - Sending email notification when CPU loading is too high (maybe above 50%). (1pt)
- You can use arbitrary solutions to do such thing, but we require you:
  - Correctly collect the metrics information and visualize them, and send mail when necessary.

Requirements – Health-Monitor

- “health-monitor”
  - **The hostname will be set to “health-monitor”.**
  - This VM will have these network interfaces:
    - Internal: To your LAN.
      - IP: Given (10.1.2.60).
  - The sshd service will be enabled.
  - Please build up a health monitoring service.
  - Sending email when web service is down. (1pt)
  - Sending email when DNS service is down. (1pt)
  - Provides dashboard for health status. (1pt)

Help Me!

- TA office hours: 15:30~17:20 Wed. at EC 324 (PC Lab).
  - We do not allow walk-ins except TA office hours or e-mail appointments.
- Questions about this homework.
  1. Make sure you have studied through lecture slides and the HW spec.
  2. Clarify your problems and google it to find out solutions.
  3. Ask them on https://groups.google.com/g/nctunasa .
     - Be sure to include all information you think others would need.
- We MIGHT give out hints on google group.
  - Be sure to join the group!
- Do not mail us unless it's personal or you're making an appointment.

Appendix – NYCU-ME

https://github.com/NYCU-ME https://nycu.me Supported By NYCU Software Development Club(SDC)

Appendix - VPN Solutions

Appendix – Health Monitoring (1)

https://linuxblog.io/50-top-server-monitoring-application-performance-monitoring-apm-solutions/

Appendix – Health Monitoring (2)

- Nagios
  - Nagios operates by checking the status of specified network services (such as HTTP, SMTP, FTP, SSH, DNS) and host resources (processor load, disk usage, system logs) at regular intervals. It can alert technical staff of problems and failures via email, SMS, or custom scripts, ensuring that any issues can be addressed promptly to minimize downtime and service disruptions.

Appendix – Health Monitoring (3)

- Nagios
  - **Monitoring Capabilities:** It can monitor network services, host resources, and environmental factors like temperature.
  - **Alerting System:** Nagios sends notifications when problems are detected and when they are resolved. Alerts can be escalated if not acknowledged within a certain timeframe.
  - **Extensibility:** Through plugins and add-ons, Nagios can monitor almost any type of infrastructure component or application.
  - **Web Interface:** It provides a web interface for viewing current network status, notification and problem history, log files, etc.
  - **Reporting:** It offers performance and availability reports, which can help in planning upgrades and identifying recurring problems.

Good Luck!

Hope this project is interesting and helps you guys score high :)