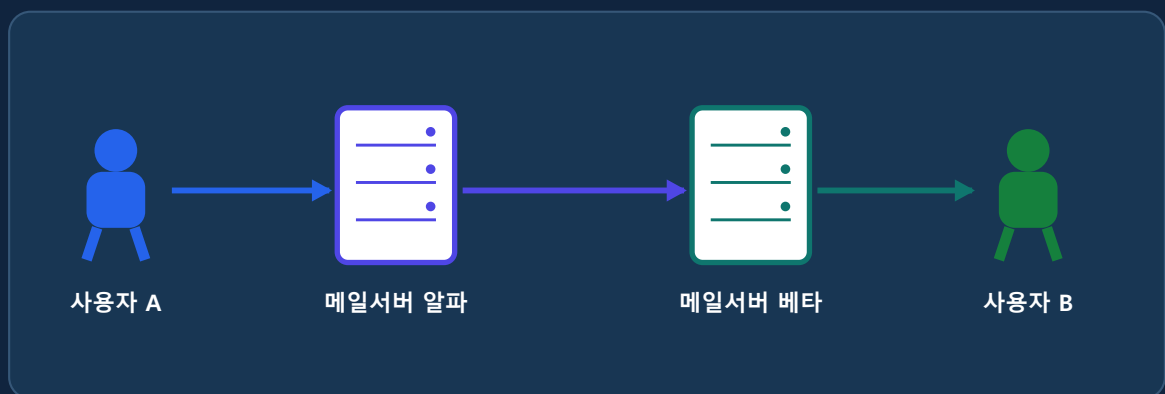


현대 메일 시스템을 그림으로 먼저 이해하는 구축 실습

Rocky Linux 9 + GNS3 + BIND + Postfix + Dovecot



처음에는 사람과 이동 경로만, 그 다음에 기술 이름을 붙입니다.

대상	메일 시스템을 처음 접하는 학습자와 강사
실습	companyA.local <-> companyB.local 양방향 송수신
원칙	그림 -> 쉬운 말 -> 기술 용어 -> 설정 -> 검증
범위	TCP는 기반이라는 점만 언급하고 메일 기능에 집중

목차

0. 기술 이름보다 먼저 큰 그림 잡기	5
1. 전문 용어 없이 보는 메일 한 통	6
1.1 다섯 장면을 말로 풀어보기	6
2. 같은 그림에 기술 이름 붙이기	7
2.1 이름표를 한 줄씩 읽기	7
3. 메일은 어느 지점에 존재하는가	8
3.1 queue와 Maildir를 구분하기	8
4. 메일 주소를 읽는 법	9
4.1 이름 관련 네 단어	9
5. 보내는 길과 읽는 길을 먼저 구분하기	10
5.1 역할과 포트	10
6. SMTP 대화는 접수 절차다	11
6.1 주요 명령의 기능	11
6.2 실제 SMTP 대화 예	12
6.3 envelope와 message header	12
7. POP3와 IMAP은 저장된 메일을 읽는다	13
7.1 POP3 주요 명령	13
7.2 IMAP 주요 명령	14
8. Postfix, Dovecot, BIND는 서로 다른 일을 한다	15
8.1 사람/서버 역할의 정식 이름	15
8.2 Postman과 Postfix	15
9. 실습 토폴로지	16
9.1 장비 표	16
9.2 구축 전에 통과해야 할 세 가지	16
9.3 관찰 도구는 각각 보는 지점이 다르다	17
10. DNS는 메일 주소의 오른쪽을 길 안내한다	18
10.1 메일 실습에 필요한 DNS 레코드	18
11. 두 BIND가 상대 도메인 질문을 넘기는 방법	19
11.1 zone 선언을 읽는 법	19

11.2 BIND 설치와 파일 준비	19
12. 메일서버 알파의 BIND 설정	20
12.1 /etc/named.conf 공통 options	20
12.2 알파의 local zone과 remote forward zone	21
12.3 알파의 forward zone 파일	21
12.4 알파의 reverse zone 파일	22
12.5 권한과 SELinux 문맥	22
13. 메일서버 베타의 BIND 설정	23
13.1 /etc/named.conf 공통 options	23
13.2 베타의 local zone과 remote forward zone	24
13.3 베타의 forward/reverse zone 파일	24
13.4 베타의 권한 설정	25
13.5 문법 검사 후 서비스 시작	25
14. DNS 검증은 질문의 의미를 알고 수행한다	26
14.1 알파에서 자기 zone과 상대 zone 확인	26
14.2 클라이언트에서 서버를 직접 지정	26
15. Postfix는 주소를 보고 두 갈래로 결정한다	27
15.1 main.cf 핵심 이름을 먼저 이해하기	27
16. 사용자 계정과 표시 주소 만들기	28
16.1 설치와 계정	28
16.2 alias가 하는 일	28
17. 메일서버 알파의 Postfix 설정	29
17.1 /etc/postfix/main.cf	29
17.2 /etc/postfix/virtual과 시작	29
18. 메일서버 베타의 Postfix 설정	30
18.1 /etc/postfix/main.cf	30
18.2 /etc/postfix/virtual과 시작	30
18.3 relay 허용과 로컬 수신을 구분하기	30
19. Maildir는 세 개의 방으로 나뉜다	31
19.1 경로 일치가 중요한 이유	31
20. Dovecot은 우편함을 열어 주는 창구다	32
20.1 설치와 공통 설정	32
20.2 인증 이름과 평문 실습 설정	33
20.3 검사와 시작	33
21. firewalld와 SELinux는 서로 다른 문이다	34

21.1 firewallld	34
21.2 SELinux	35
22. 한 통의 메일을 8개 번호로 추적한다	36
22.1 시작 전 6개 질문	36
23. swaks로 전체 발신 경로를 한 번에 검증	37
23.1 출력에서 순서대로 찾을 것	37
24. nc로 SMTP 명령을 직접 말해 보기	38
24.1 빈 줄과 단독 점은 서로 다르다	38
25. 베타의 디스크에서 저장 사실 확인	39
25.1 파일이 없을 때 볼 것	39
26. Client B에서 IMAP으로 우편함 열기	40
26.1 Maildir에는 있지만 0 EXISTS라면	40
27. POP3로 같은 메시지를 확인하고 반대 방향도 시험	41
27.1 Client B의 POP3 확인	41
27.2 사용자 B -> 사용자 A 반대 방향	41
28. queue ID는 메시지의 배송 조회 번호다	42
28.1 로그와 queue 명령	42
28.2 Postfix 로그 한 줄 번역	43
28.3 애플리케이션 대화 캡처	43
29. 증상보다 먼저 멈춘 단계를 찾는다	44
29.1 초심자용 증상 표	44
30. 고의로 한 곳만 망가뜨려 배우기	45
30.1 복구 원칙	45
31. Dovecot LMTP와 TLS 확장	46
31.1 Dovecot을 실제 MDA로 만들기	46
31.2 TLS는 무엇을 바꾸는가	47
32. 강의 진행 순서와 완료 체크리스트	48
32.1 설명 순서	48
32.2 설명 체크	48
32.3 구축 체크	48
33. 한 장 용어 요약	49
34. 공식 참고 자료	50

먼저 볼 것

등장인물과 메일 한 통의 여행

0. 기술 이름보다 먼저 큰 그림 잡기

이 장에서는 SMTP, MTA 같은 약어를 외우지 않는다. 먼저 누가 무엇을 하고, 메일이 어느 순서로 이동하며, 마지막에는 어디에 저장되는지만 본다.

그림 0-1. 이 실습에 등장하는 네 주체



알파와 베타는 서로 다른 회사의 메일서버다. 한 장비 안에서 여러 프로그램이 협력하지만, 지금은 각각 하나의 우체국처럼 생각하면 된다.

한 문장으로 먼저 기억하기

사용자 A가 알파에 메일을 맡기면, 알파가 베타를 찾아 전달하고, 베타는 사용자 B의 개인 우편함에 저장한다. 사용자 B는 나중에 그 우편함을 열어 읽는다.

1. 전문 용어 없이 보는 메일 한 통

메일은 한 번에 사용자 A의 화면에서 사용자 B의 화면으로 날아가지 않는다. 접수, 길 찾기, 서버 간 전달, 저장, 열람이라는 다섯 장면을 거친다.

그림 1-1. 사용자 A가 쓴 메일이 사용자 B에게 보일 때까지

먼저 이 다섯 장면만 이해한다. 기술 이름은 다음 그림에서 붙인다.



이 그림의 핵심은 '서버가 메일을 저장한다'는 점이다. 사용자 B가 자리에 없어도 베타가 우편함에 보관하므로 나중에 읽을 수 있다.

1.1 다섯 장면을 말로 풀어보기

1. 사용자 A가 받는 사람 주소와 내용을 적는다.
2. 메일서버 알파가 메일을 접수하고 추적할 수 있는 대기 번호를 붙인다.
3. 알파가 주소 안내 서비스에 물어 메일서버 베타의 위치를 찾는다.
4. 알파가 베타에게 메일을 넘기고, 베타가 사용자 B의 개인 우편함에 저장한다.
5. 사용자 B가 자기 우편함에 접속해 도착한 메일을 읽는다.

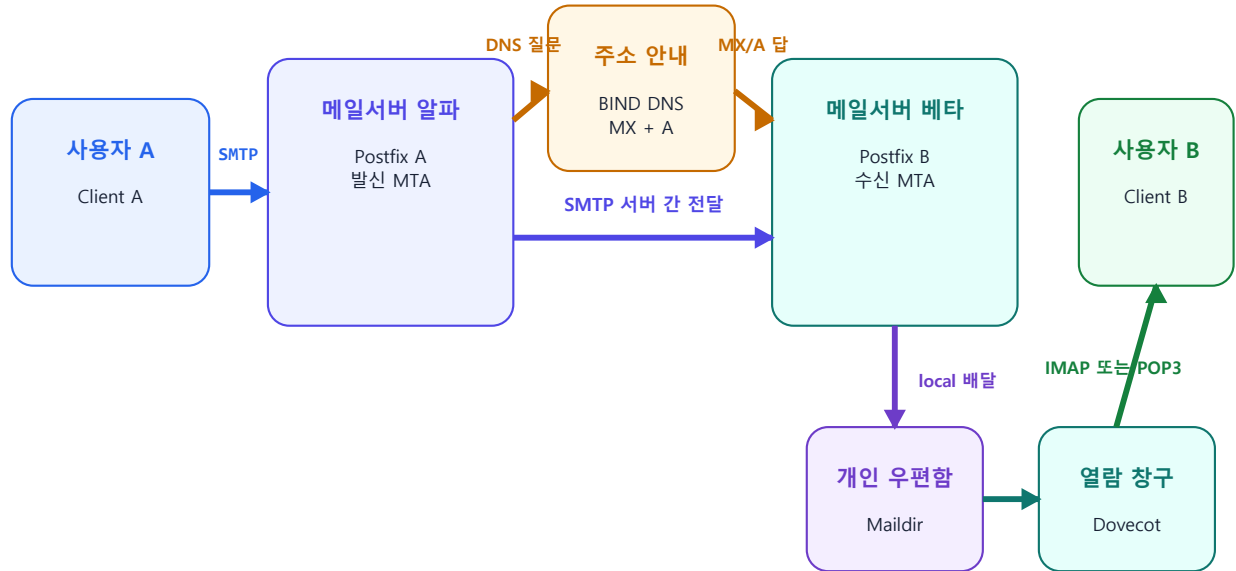
여기서 아직 외우지 않아도 되는 것

각 장면의 기술 이름과 포트 번호는 다음 장에서 같은 그림 위에 붙인다. 처음부터 약어를 외우기보다 먼저 '무슨 일인가'를 설명할 수 있어야 한다.

2. 같은 그림에 기술 이름 붙이기

이제 앞에서 본 다섯 장면을 그대로 두고, 실제 프로그램과 프로토콜 이름을 붙인다. 새 개념을 추가하는 것이 아니라 이미 이해한 역할에 이름표를 붙이는 단계다.

그림 2-1. 쉬운 역할과 실제 기술 이름의 대응



핵심: 보내는 길은 SMTP, 저장된 메일을 읽는 길은 IMAP 또는 POP3다.

SMTP, DNS, Postfix, Maildir, Dovecot, IMAP/POP3는 서로 경쟁하는 한 종류의 프로그램이 아니다. 각자 맡은 장면이 다르다.

2.1 이름표를 한 줄씩 읽기

앞 그림의 쉬운 표현	기술 이름	이 실습에서 하는 일
메일을 맡김	SMTP	Client A가 Postfix A에 새 메일을 제출
받을 곳을 찾음	DNS의 MX와 A 레코드	companyB.local 담당 서버 이름과 IP 확인
서버끼리 넘김	SMTP	Postfix A가 Postfix B의 25번 포트로 전달
개인 우편함에 넣음	Postfix local(8) + Maildir	userb의 홈 디렉터리에 파일로 저장
우편함을 열어 읽음	Dovecot + IMAP 또는 POP3	Client B에게 목록과 본문 제공

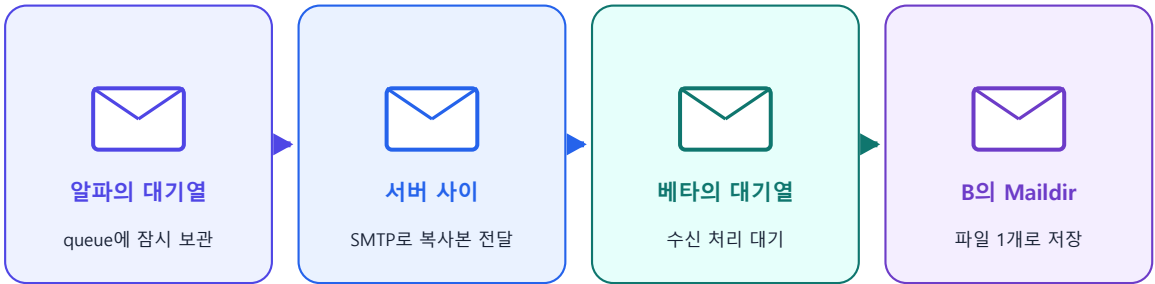
TCP는 어디에 있는가

SMTP, POP3, IMAP은 모두 TCP를 기반으로 동작한다. 이 교재는 TCP 연결 성립 과정은 이미 안다고 가정하고, 연결된 뒤 메일 프로그램이 어떤 명령과 응답을 주고받는지에 집중한다.

3. 메일은 어느 지점에 존재하는가

'전송 중인 메일'도 실제로는 각 서버의 저장 공간을 잠시 거친다. 이 사실을 알면 로그와 장애 진단이 쉬워진다.

그림 3-1. 메시지가 거치는 저장 지점



메일은 순간이동하지 않는다. 각 서버가 받아서 저장하고 다음 단계에 넘긴다.
그래서 오류가 나면 어느 칸까지 왔는지 확인하면 된다.

queue는 아직 다음 처리가 끝나지 않은 메시지의 대기 목록이다. Maildir는 최종 수신자의 메일을 장기적으로 저장하는 개인 우편함이다.

3.1 queue와 Maildir를 구분하기

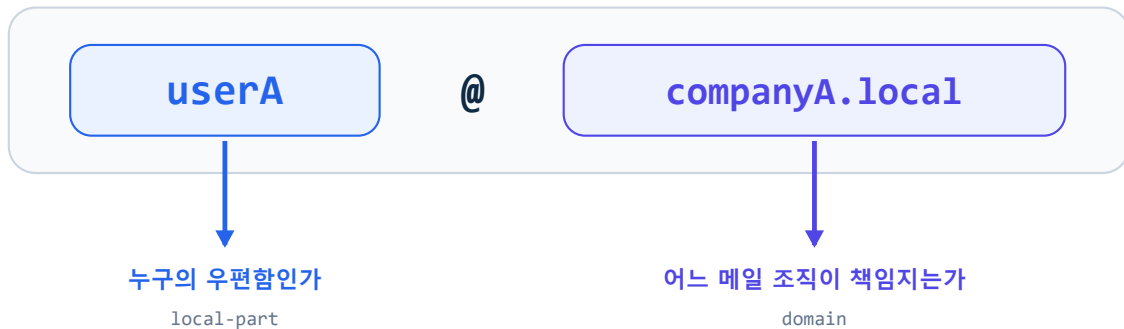
단어	쉬운 뜻	언제 보이는가	확인 방법
queue	전달을 기다리거나 재시도 중인 메일 목록	알파/베타가 메시지를 처리하는 동안	postqueue -p
queue ID	서버 안에서 메시지를 추적하는 번호	250 queued 응답과 Postfix 로그	journalctl -u postfix
Maildir	수신자별 실제 우편함 디렉터리	베타가 userb에게 최종 배달한 뒤	/home/userb/Maildir/new
mailbox	사용자의 수신 메시지가 모이는 논리 공간	IMAP의 INBOX 같은 화면	SELECT INBOX

장애를 좁히는 질문
알파의 queue에도 없다면 접수 전 문제다. 알파에는 있고 베타에는 없다면 서버 간 전달 문제다. 베타의 Maildir에는 있지만 사용자 화면에 없다면 Dovecot 또는 로그인 문제다.

4. 메일 주소를 읽는 법

주소는 '누구에게'와 '어느 메일 조직에'를 한 줄에 적는다. SMTP 서버는 특히 오른쪽 domain을 보고 로컬 배달인지 다른 서버로 전달할지 결정한다.

그림 4-1. userA@companyA.local 주소 해부



이메일의 @는 주소를 두 부분으로 나누는 기호다. BIND zone 파일의 @는 현재 zone 자체를 뜻하므로 의미가 다르다.

4.1 이름 관련 네 단어

단어	처음 보는 사람을 위한 설명	예
local-part	@ 왼쪽. 해당 도메인 안에서 어느 사용자 우편함인지 가리킴	userA, userB
domain	@ 오른쪽. 어느 메일 체계가 이 주소를 책임지는지 나타냄	companyA.local
hostname	네트워크의 특정 장비나 서비스를 가리키는 이름	mail.companyA.local
FQDN	오른쪽 끝까지 모두 적은 완전한 DNS 이름. zone 파일에서는 끝 점까지 적음	mail.companyB.local.

.local 접미사 주의

일부 데스크톱은 .local을 mDNS용으로 특별 취급한다. 요청한 실습 도메인은 유지하지만 검사할 때는 `dig @192.168.10.10 ...`처럼 DNS 서버를 직접 지정한다. 실제 조직에서는 소유한 도메인의 하위 도메인이나 교육용 .test 사용을 권장한다.

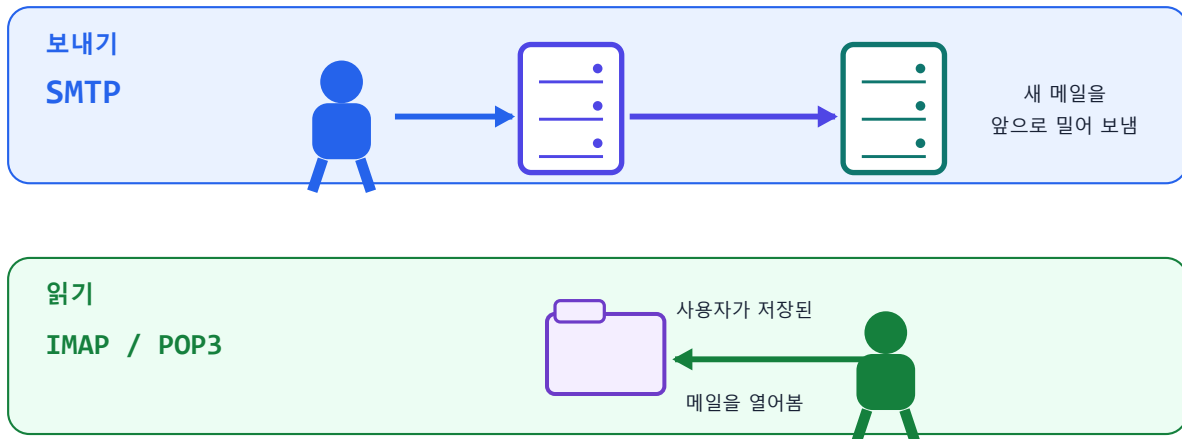
PART 1

메일 프로토콜은 무엇을 하는가

5. 보내는 길과 읽는 길을 먼저 구분하기

메일 프로토콜은 모두 같은 일을 하지 않는다. 새 메일을 앞으로 보내는 길과, 이미 저장된 메일을 사용자가 읽는 길이 나뉜다.

그림 5-1. SMTP와 IMAP/POP3의 방향



SMTP는 새 메시지를 다음 서버로 밀어 보낸다. IMAP과 POP3는 서버에 저장된 메시지를 사용자가 조회하는 데 쓴다.

5.1 역할과 포트

서비스	기본 포트	암호화 방식	주 용도	기억 문장
SMTP relay	25	평문 시작 후 STARTTLS 가능	메일 서버끼리 전달	MX 배달은 25
Submission	587	STARTTLS	사용자가 인증 후 제출	운영 MUA 발신
Submissions	465	처음부터 TLS	사용자 암호화 제출	운영 MUA 발신
POP3	110	평문 또는 STLS	목록과 메시지 다운로드	단순 수신
POP3S	995	처음부터 TLS	암호화 POP3	운영 수신
IMAP	143	평문 또는 STARTTLS	서버 우편함 동기화	폴더/상태 관리
IMAPS	993	처음부터 TLS	암호화 IMAP	운영 수신

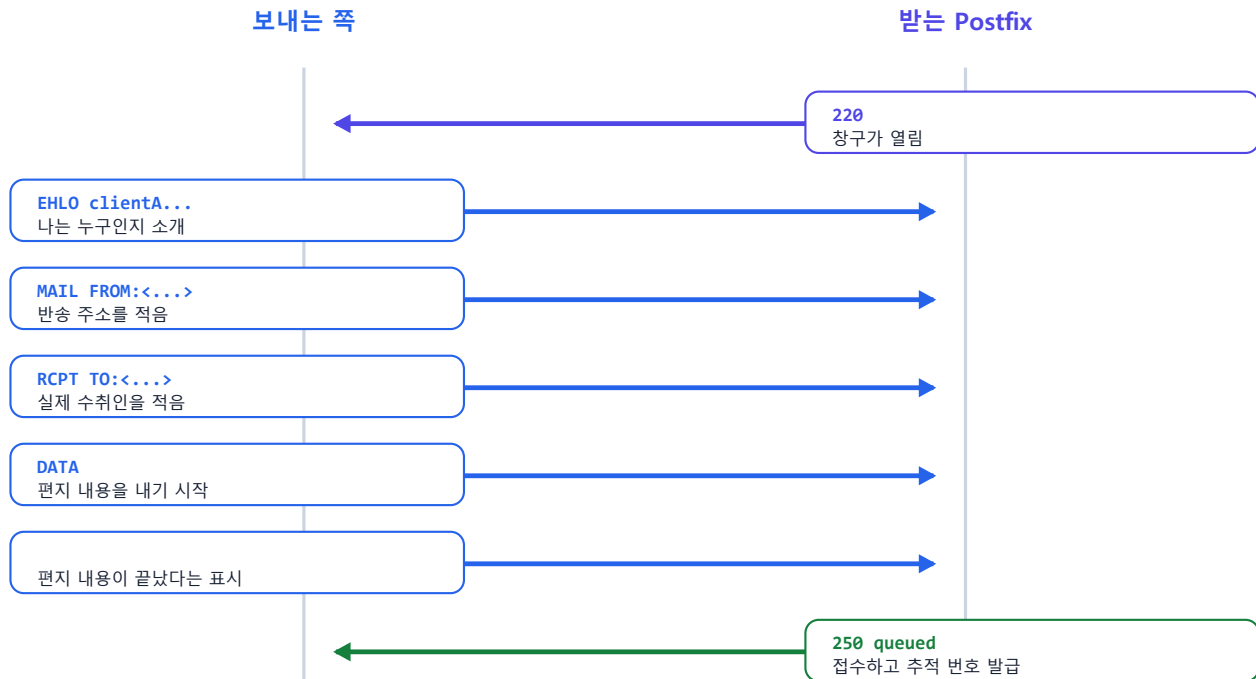
이 실습이 평문 25/110/143을 쓰는 이유

완전히 격리된 교육망에서 명령과 응답을 눈으로 관찰하기 위해서다. 실제 사용자 환경에서는 인증서와 TLS를 구성하고 587/465, 993, 995 사용을 권장한다.

6. SMTP 대화는 접수 절차다

SMTP 연결 뒤에는 보내는 쪽이 순서대로 신원, 반송 주소, 수취인, 메시지 내용을 말한다. 받는 서버는 각 단계에 숫자 응답을 돌려준다.

그림 6-1. SMTP 명령과 응답을 접수 창구처럼 읽기



숫자 2xx는 성공 계열이다. 4xx는 일시 문제가 재시도할 수 있고, 5xx는 주소나 권한 같은 영구 실패 계열이다.

6.1 주요 명령의 기능

명령/응답	서버가 이해하는 의미	처음 보는 사람의 비유	실패 시 먼저 볼 곳
220	SMTP 대화를 받을 준비가 됨	접수 창구가 열림	서비스/listen/firewall
EHLO	클라이언트 이름과 확장 기능 확인	어디서 왔는지 소개	이름 형식, 서버 응답
MAIL FROM	반송에 쓸 envelope sender 지정	반송 주소 적기	주소 문법/정책
RCPT TO	실제 배달할 envelope recipient 지정	수취인 주소 적기	relay denied/user unknown
DATA	헤더와 본문 입력 시작	봉투 안 편지 제출	354 응답 여부
단독 점	DATA 입력 종료	편지 입력 완료	새 줄의 점 누락
250 queued	서버가 책임을 인수하고 queue ID 발급	접수증 발급	이후 queue ID로 추적

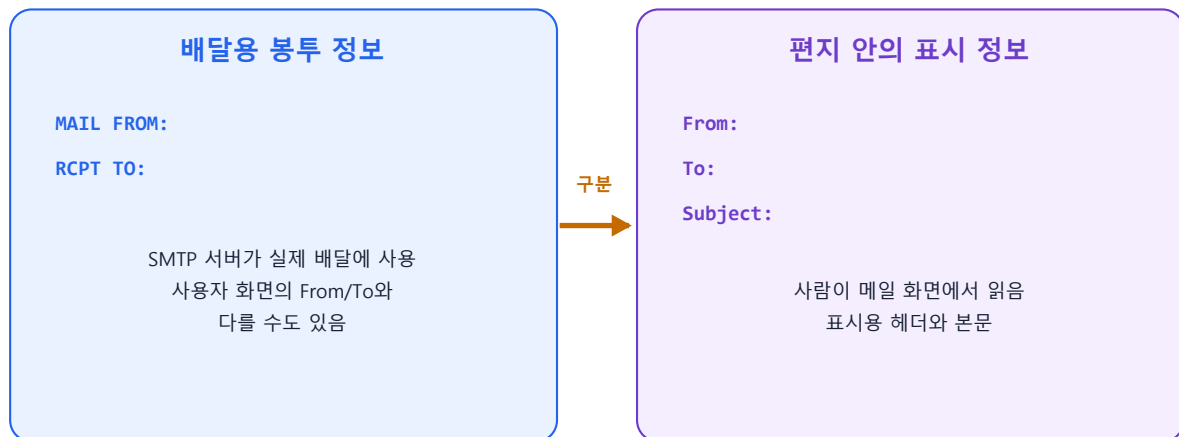
6.2 실제 SMTP 대화 예

Client A와 Postfix A 사이

```
S: 220 mail.companyA.local ESMTP Postfix
C: EHLO clientA.companyA.local
S: 250 mail.companyA.local
C: MAIL FROM:<userA@companyA.local>
S: 250 2.1.0 Ok
C: RCPT TO:<userB@companyB.local>
S: 250 2.1.5 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: From: User A <userA@companyA.local>
C: To: User B <userB@companyB.local>
C: Subject: first visual lab
C:
C: Hello User B.
C: .
S: 250 2.0.0 Ok: queued as 9A18C12345
C: QUIT
```

6.3 envelope와 message header

그림 6-2. 배달용 주소와 화면 표시용 주소



MAIL FROM/RCPT TO는 SMTP가 실제 배달에 쓰는 envelope다. DATA 안의 From/To/Subject는 사람이 메일 화면에서 읽는 header다.

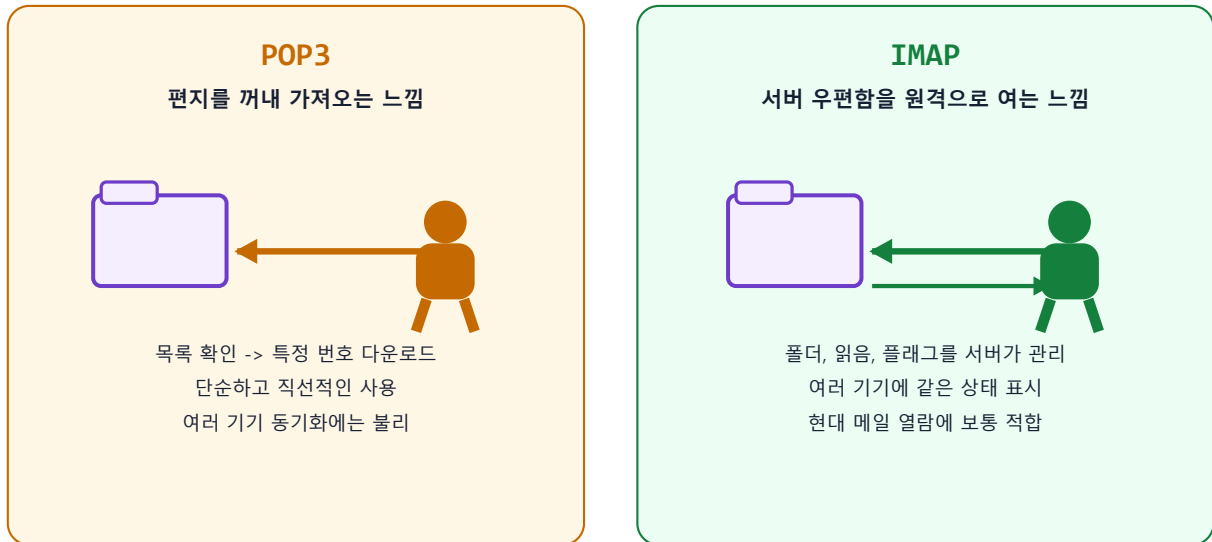
왜 구분해야 하는가

로그의 to=<...>는 실제 envelope recipient다. 화면의 To:는 표시용 header다. 스푸핑이나 반송 문제를 분석할 때 두 주소를 같다고 가정하면 원인을 놓칠 수 있다.

7. POP3와 IMAP은 저장된 메일을 읽는다

두 프로토콜 모두 Dovecot을 통해 Maildir의 메시지를 보여 주지만 사용 방식이 다르다. POP3는 단순 다운로드, IMAP은 서버 우편함 원격 관리에 가깝다.

그림 7-1. POP3와 IMAP의 사용 느낌



여러 기기에서 같은 폴더와 읽음 상태를 유지하려면 IMAP이 더 자연스럽다. POP3는 구조가 단순해 명령 학습에 좋다.

7.1 POP3 주요 명령

명령	뜻	관찰할 결과
USER / PASS	사용자 이름과 비밀번호로 인증	+OK Logged in
STAT	메시지 수와 전체 크기 확인	+OK 1 742
LIST	메시지 번호와 크기 목록	1 742
RETR 1	1번 메시지 전체 내용 받기	header와 body
DELE 1	1번을 삭제 대상으로 표시	QUIT 시 반영 가능
QUIT	정상 종료	+OK Logging out

POP3 예

```

S: +OK Dovecot ready.
C: USER userB@companyB.local
C: PASS <PASSWORD>
S: +OK Logged in.
C: STAT
S: +OK 1 742
C: RETR 1
S: +OK 742 octets
S: [message]
C: QUIT
  
```

7.2 IMAP 주요 명령

IMAP 명령 앞의 a001, a002 같은 표시는 tag다. 클라이언트가 보낸 명령과 서버의 완료 응답을 짝지어 찾는 번호표라고 생각하면 된다.

명령	뜻	상태 변화
CAPABILITY	서버가 지원하는 기능 확인	연결 직후
LOGIN	사용자 인증	Not Authenticated -> Authenticated
SELECT INBOX	받은편지함 선택	Authenticated -> Selected
FETCH	선택한 메시지의 헤더/본문/플래그 조회	Selected 상태에서 사용
STORE	읽음/삭제 등의 플래그 변경	Selected 상태에서 사용
LOGOUT	세션 종료	Logout

IMAP 예

```
S: * OK Dovecot ready.
C: a001 LOGIN userB@companyB.local "<PASSWORD>"
S: a001 OK Logged in.
C: a002 SELECT INBOX
S: * 1 EXISTS
S: a002 OK [READ-WRITE] Select completed.
C: a003 FETCH 1 BODY.PEEK[]
S: * 1 FETCH (...)
S: a003 OK Fetch completed.
C: a004 LOGOUT
```

BODY.PEEK[]의 의미

BODY[]는 메시지 본문을 읽고 읽음 상태가 바뀔 수 있다. BODY.PEEK[]는 내용을 보되 불필요하게 읽음 플래그를 바꾸지 않도록 요청한다.

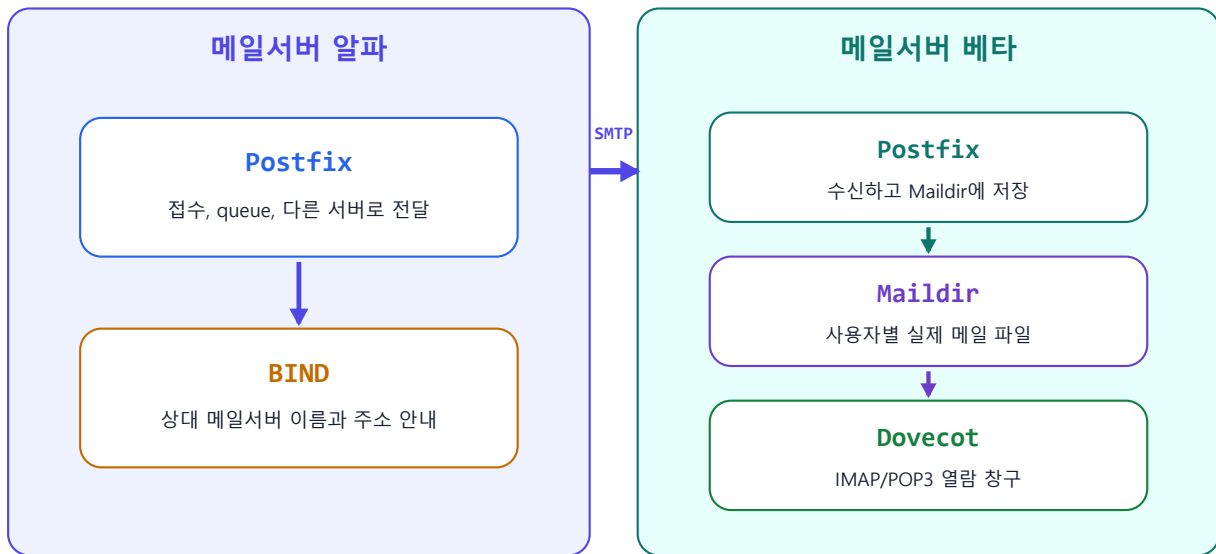
PART 2

한 서버 안에서 프로그램들이 나누는 역할

8. Postfix, Dovecot, BIND는 서로 다른 일을 한다

메일서버라는 장비 하나 안에서 여러 프로그램이 협력한다. 프로그램 이름보다 먼저 각 프로그램이 맡은 장면을 그림으로 연결한다.

그림 8-1. 알파와 베타 안의 프로그램 역할



알파와 베타 모두 실제로는 BIND, Postfix, Dovecot을 설치한다. 그림은 한 통의 진행 방향을 강조하기 위해 주 역할만 표시했다.

8.1 사람/서버 역할의 정식 이름

분류	한 문장 정의	우체국 비유	대표 프로그램/실습 역할
MUA	사용자가 메일을 쓰고 읽는 프로그램	편지를 쓰는 책상	Thunderbird, swaks, nc
MSA	인증된 사용자의 발신 메일을 접수	고객용 접수 창구	운영에서는 587/465, 기준 실습은 생략
MTA	메일을 queue에 두고 다른 서버로 전달	분류 센터와 운송 차량	Postfix, Sendmail
MDA/LDA	최종 수신자의 사서함에 넣는 구성요소	마지막 배달 담당자	기준: Postfix local(8), 확장: Dovecot LMTP
MRA	저장된 메일을 IMAP/POP3로 제공	우편함 열람 창구	Dovecot
Webmail	브라우저 화면을 제공하는 MUA	웹 접수/열람 창구	Roundcube, 내부적으로 IMAP/SMTP 사용

Dovecot을 무조건 MDA라고 부르면 생기는 혼동

이 교재의 기준 설정 `home_mailbox = Maildir/`에서는 Postfix의 `local(8)`이 파일을 만든다. Dovecot은 그 파일을 읽어 IMAP/POP3로 제공하므로 우선 MRA다. LMTP 배달까지 Dovecot에 맡기는 확장 구성에서는 MDA 역할도 수행한다.

8.2 Postman과 Postfix

이름만 비슷하다

Postman은 HTTP API 테스트 도구이며 메일을 queue에 두거나 MX를 조회하지 않는다. Postfix는 SMTP MTA다. 비유하면 Postman은 웹 서비스용 검사 도구이고 Postfix는 실제 우편 분류 센터다.

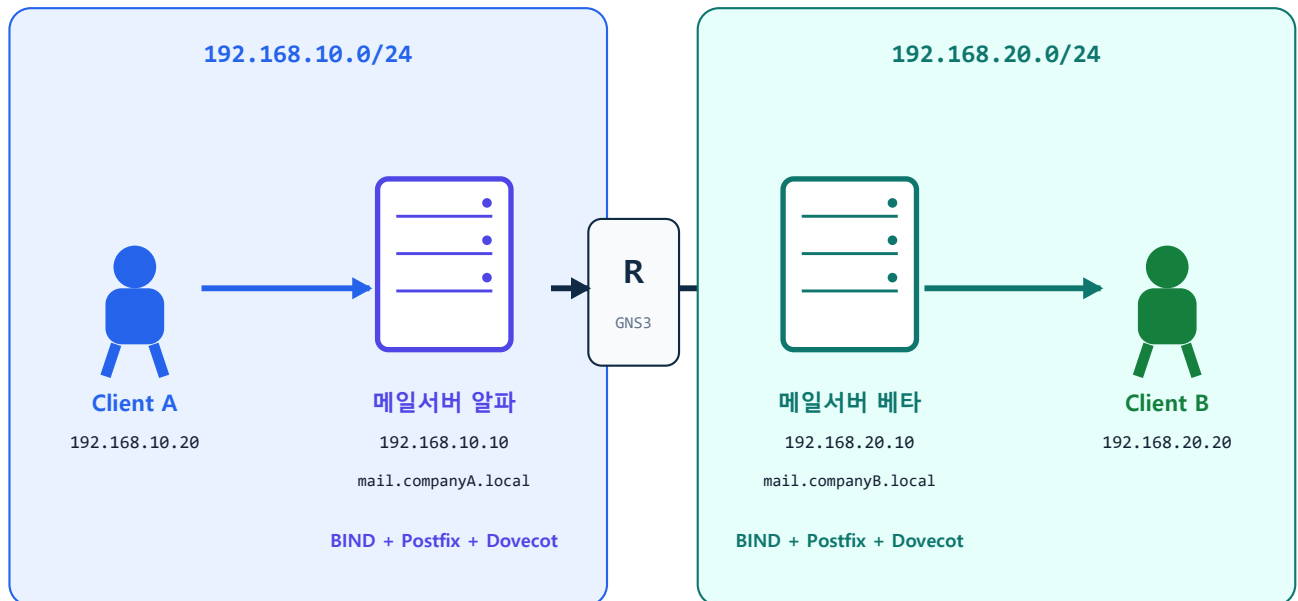
PART 3

GNS3 폐쇄망에 두 회사를 배치하기

9. 실습 토폴로지

외부 인터넷과 단절된 두 서버넷을 GNS3 라우터로 연결한다. 알파와 베타는 각각 자기 회사의 DNS와 메일 서비스를 한 장비에서 제공한다.

그림 9-1. 주소와 역할이 표시된 전체 토폴로지



라우터 인터페이스 예시는 192.168.10.1과 192.168.20.1이다. 두 네트워크 사이에 양방향 라우팅이 되어야 한다.

9.1 장비 표

장비	IP	이름/도메인	역할
메일서버 알파	192.168.10.10/24	mail.companyA.local	BIND, Postfix, Dovecot
Client A	192.168.10.20/24	clientA.companyA.local	사용자 A 송신
메일서버 베타	192.168.20.10/24	mail.companyB.local	BIND, Postfix, Dovecot
Client B	192.168.20.20/24	clientB.companyB.local	사용자 B 수신

9.2 구축 전에 통과해야 할 세 가지

1. Client A와 Client B가 각각 자기 게이트웨이를 통과해 반대편 서버 IP로 ping할 수 있다.
2. 서버 이름과 IP를 혼동하지 않는다. MX는 IP가 아니라 메일서버 이름을 가리킨다.
3. 폐쇄망에 필요한 RPM이 준비되어 있다. swaks가 없으면 nc로 같은 실습을 진행한다.

두 서버와 클라이언트의 네트워크 확인

```
ip -br address
ip route
ping -c 3 192.168.10.1
ping -c 3 192.168.20.1
ping -c 3 192.168.20.10
```

9.3 관찰 도구는 각각 보는 지점이 다르다

Client 진단 도구

```
dnf install -y bind-utils nmap-ncat tcpdump telnet

# Optional: EPEL package
dnf install -y swaks
```

도구	무엇을 보여 주는가	어느 단계에 쓰는가
dig / nslookup	DNS의 MX, A, PTR 응답	담당 서버와 IP 찾기
swaks	SMTP 대화를 자동으로 만들고 응답을 표시	전체 발신 경로 빠른 확인
nc / telnet	명령을 한 줄씩 직접 입력	SMTP, IMAP, POP3 기능 학습
tcpdump / Wireshark	실제 애플리케이션 대화 캡처	평문 명령과 메시지 관찰

폐쇄망 패키지 준비

인터넷이 끊긴 뒤 설치를 시작하면 필요한 패키지를 받을 수 없다. 실습 이미지에 RPM과 의존성을 미리 넣거나 내부 저장소를 준비한다. swaks가 없어도 nc 실습은 진행할 수 있다.

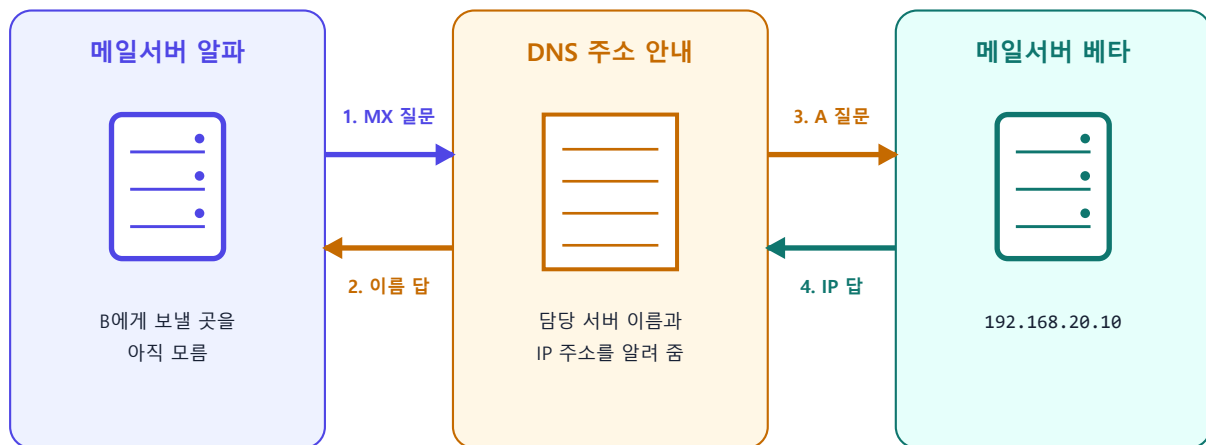
PART 4

DNS가 담당 메일서버를 알려 주는 과정

10. DNS는 메일 주소의 오른쪽을 길 안내한다

Postfix A는 userB@companyB.local을 보고 companyB.local의 담당 메일서버를 알아야 한다. DNS는 먼저 담당 서버의 이름을, 이어서 그 이름의 IP 주소를 알려 준다.

그림 10-1. MX와 A를 차례로 찾는 이유



MX는 담당 서버의 이름, A는 그 이름의 IPv4 주소를 알려 준다.

MX 레코드는 담당 메일서버의 이름을 가리킨다. 실제 연결에는 IP가 필요하므로 그 이름의 A 레코드를 다시 조회한다.

10.1 메일 실습에 필요한 DNS 레코드

레코드	쉬운 설명	우체국 비유	이 실습의 예
A	호스트 이름을 IPv4 주소로 연결	건물 이름의 실제 위치	mail.companyA.local -> 192.168.10.10
MX	도메인의 메일을 받을 서버와 우선순위	지역 담당 우체국	companyB.local -> mail.companyB.local
PTR	IP 주소에서 이름을 찾는 역방향 정보	위치에서 건물 이름 찾기	192.168.20.10 -> mail.companyB.local
NS	zone의 권한 DNS 서버	주소록 원본 관리자	ns1.companyA.local
SOA	zone의 관리 정보와 serial	주소록 표지와 개정 번호	2026082401
TTL	응답을 캐시에 유지할 시간	안내지를 다시 확인할 주기	300초

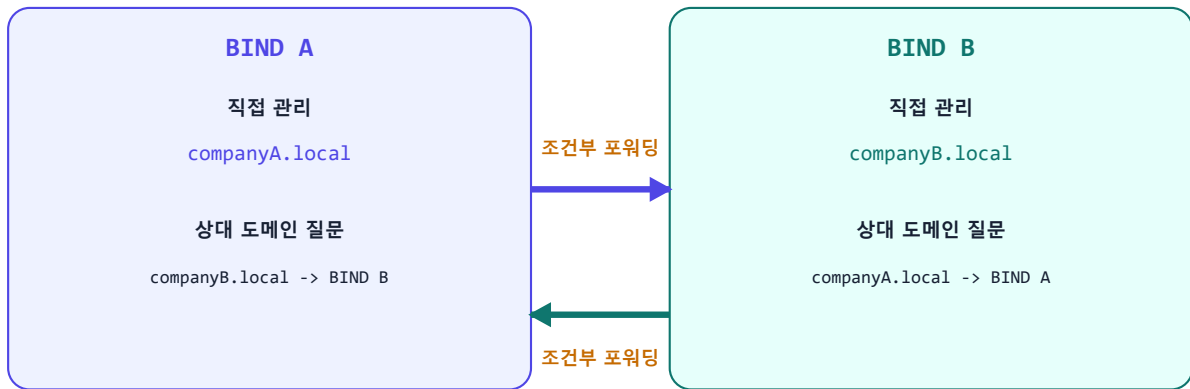
MX target 끝의 점

zone 파일에서 mail.companyA.local.처럼 마지막 점을 붙이면 완전한 이름이다. 점을 빼면 현재 zone 이름이 다시 붙어 mail.companyA.local.companyA.local이 될 수 있다.

11. 두 BIND가 상대 도메인 질문을 넘기는 방법

각 BIND는 자기 회사 zone의 원본만 관리한다. 상대 회사에 관한 질문이 오면 지정한 상대 BIND로만 전달한다. 이를 조건부 포워딩이라고 한다.

그림 11-1. 양방향 조건부 포워딩



A는 companyA.local에 직접 답하고 companyB.local 질문만 B에게 전달한다. B는 반대로 동작한다.

11.1 zone 선언을 읽는 법

설정	쉬운 뜻
type primary	이 서버가 해당 zone 파일의 원본 관리자
type forward	해당 zone 질문을 다른 DNS 서버로 전달
forward only	지정한 forwarder에게만 묻고 다른 경로로 우회하지 않음
allow-query	어떤 client가 DNS 질문을 할 수 있는지 제한
allow-recursion	어떤 client를 대신해 다른 DNS에 물어볼 수 있는지 제한
allow-transfer none	교육망에서 불필요한 zone 전체 복사를 차단

11.2 BIND 설치와 파일 준비

무엇을 하는가	왜 필요한가	정상이라면
bind를 설치하고 named를 아직 시작하지 않은 상태에서 설정 파일을 작성한다.	잘못된 설정으로 서비스가 반복 재시작되는 상황을 피하고 먼저 문법을 검사하기 위해서다.	named-checkconf와 named-checkzone이 아무 오류 없이 끝난다.

두 서버

```
dnf install -y bind bind-utils
named -v
```

12. 메일서버 알파의 BIND 설정

12.1 /etc/named.conf 공통 options

알파는 192.168.10.10에서 DNS 질문을 받고 두 실습 서브넷의 질문과 재귀를 허용한다. 인터넷 전체에 공개하는 설정이 아니라 폐쇄망 범위만 명시한다.

메일서버 알파 - /etc/named.conf 앞부분

```
acl lab_nets {
    127.0.0.1;
    192.168.10.0/24;
    192.168.20.0/24;
};

options {
    listen-on port 53 { 127.0.0.1; 192.168.10.10; };
    listen-on-v6 port 53 { none; };
    directory "/var/named";
    dump-file "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    secroots-file "/var/named/data/named.secroots";
    recursing-file "/var/named/data/named.recursing";

    allow-query { lab_nets; };
    allow-recursion { lab_nets; };
    allow-transfer { none; };
    recursion yes;
    dnssec-validation yes;
    minimal-responses yes;

    managed-keys-directory "/var/named/dynamic";
    pid-file "/run/named/named.pid";
    session-keyfile "/run/named/session.key";
    include "/etc/crypto-policies/back-ends/bind.config";
};

zone "." IN {
    type hint;
    file "named.ca";
};
```

12.2 알파의 local zone과 remote forward zone

companyA.local과 192.168.10.0/24 역방향 zone은 알파가 직접 관리한다. B 쪽 이름과 역방향 질문은 192.168.20.10으로 보낸다.

메일서버 알파 - /etc/named.conf 뒷부분

```
zone "companyA.local" IN {
    type primary;
    file "companyA.local.zone";
    allow-update { none; };
};

zone "10.168.192.in-addr.arpa" IN {
    type primary;
    file "10.168.192.rev";
    allow-update { none; };
};

zone "companyB.local" IN {
    type forward;
    forward only;
    forwarders { 192.168.20.10; };
};

zone "20.168.192.in-addr.arpa" IN {
    type forward;
    forward only;
    forwarders { 192.168.20.10; };
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
```

12.3 알파의 forward zone 파일

/var/named/companyA.local.zone

```
$TTL 300
@ IN SOA ns1.companyA.local. hostmaster.companyA.local. (
    2026082401
    300
    60
    604800
    300
)

IN NS ns1.companyA.local.
IN MX 10 mail.companyA.local.

ns1 IN A 192.168.10.10
mail IN A 192.168.10.10
clientA IN A 192.168.10.20
```

12.4 알파의 reverse zone 파일

`/var/named/10.168.192.rev`

```
$TTL 300
@   IN SOA ns1.companyA.local. hostmaster.companyA.local. (
      2026082401 300 60 604800 300
)

      IN NS ns1.companyA.local.
10   IN PTR mail.companyA.local.
20   IN PTR clientA.companyA.local.
```

serial은 주소록 개정 번호

zone 파일을 수정할 때 serial을 증가시키는 습관을 들인다. 이 실습에는 secondary DNS가 없지만 변경 추적과 문법 검증이 쉬워진다.

12.5 권한과 SELinux 문맥

메일서버 알파

```
chown root:named /var/named/companyA.local.zone \
    /var/named/10.168.192.rev
chmod 640 /var/named/companyA.local.zone \
    /var/named/10.168.192.rev
restorecon -v /var/named/companyA.local.zone \
    /var/named/10.168.192.rev
```

13. 메일서버 베타의 BIND 설정

베타의 구조는 알파의 거울상이다. 주소만 192.168.20.10으로 바꾸고, companyB.local을 직접 관리하며 companyA.local 질문을 알파로 전달한다.

13.1 /etc/named.conf 공통 options

메일서버 베타 - /etc/named.conf 앞부분

```
acl lab_nets {
    127.0.0.1;
    192.168.10.0/24;
    192.168.20.0/24;
};

options {
    listen-on port 53 { 127.0.0.1; 192.168.20.10; };
    listen-on-v6 port 53 { none; };
    directory "/var/named";
    dump-file "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    secroots-file "/var/named/data/named.secroots";
    recursing-file "/var/named/data/named.recursing";

    allow-query { lab_nets; };
    allow-recursion { lab_nets; };
    allow-transfer { none; };
    recursion yes;
    dnssec-validation yes;
    minimal-responses yes;

    managed-keys-directory "/var/named/dynamic";
    pid-file "/run/named/named.pid";
    session-keyfile "/run/named/session.key";
    include "/etc/crypto-policies/back-ends/bind.config";
};

zone "." IN {
    type hint;
    file "named.ca";
};
```

13.2 베타의 local zone과 remote forward zone

메일서버 베타 - /etc/named.conf 뒷부분

```
zone "companyB.local" IN {
    type primary;
    file "companyB.local.zone";
    allow-update { none; };
};

zone "20.168.192.in-addr.arpa" IN {
    type primary;
    file "20.168.192.rev";
    allow-update { none; };
};

zone "companyA.local" IN {
    type forward;
    forward only;
    forwarders { 192.168.10.10; };
};

zone "10.168.192.in-addr.arpa" IN {
    type forward;
    forward only;
    forwarders { 192.168.10.10; };
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
```

13.3 베타의 forward/reverse zone 파일

/var/named/companyB.local.zone

```
$TTL 300
@ IN SOA ns1.companyB.local. hostmaster.companyB.local. (
    2026082401 300 60 604800 300
)

IN NS ns1.companyB.local.
IN MX 10 mail.companyB.local.

ns1 IN A 192.168.20.10
mail IN A 192.168.20.10
clientB IN A 192.168.20.20
```

/var/named/20.168.192.rev

```
$TTL 300
@ IN SOA ns1.companyB.local. hostmaster.companyB.local. (
    2026082401 300 60 604800 300
)

IN NS ns1.companyB.local.
10 IN PTR mail.companyB.local.
20 IN PTR clientB.companyB.local.
```

13.4 베타의 권한 설정

메일서버 베타

```
chown root:named /var/named/companyB.local.zone \
    /var/named/20.168.192.rev
chmod 640 /var/named/companyB.local.zone \
    /var/named/20.168.192.rev
restorecon -v /var/named/companyB.local.zone \
    /var/named/20.168.192.rev
```

13.5 문법 검사 후 서비스 시작

검사 명령이 성공하면 아무 메시지 없이 종료되거나 zone loaded/OK가 보인다. 오류 줄 번호가 나오면 서비스를 시작하기 전에 해당 파일을 고친다.

메일서버 알파

```
named-checkconf
named-checkzone companyA.local /var/named/companyA.local.zone
named-checkzone 10.168.192.in-addr.arpa /var/named/10.168.192.rev
systemctl enable --now named
systemctl --no-pager --full status named
```

메일서버 베타

```
named-checkconf
named-checkzone companyB.local /var/named/companyB.local.zone
named-checkzone 20.168.192.in-addr.arpa /var/named/20.168.192.rev
systemctl enable --now named
systemctl --no-pager --full status named
```

14. DNS 검증은 질문의 의미를 알고 수행한다

14.1 알파에서 자기 zone과 상대 zone 확인

메일서버 알파

```
dig @127.0.0.1 companyA.local MX +short
dig @127.0.0.1 mail.companyA.local A +short
dig @127.0.0.1 companyB.local MX +short
dig @127.0.0.1 mail.companyB.local A +short
dig @127.0.0.1 -x 192.168.20.10 +short
```

질문	왜 묻는가	기대 결과
companyA.local MX	알파가 자기 도메인 담당 서버를 직접 답하는가	10 mail.companyA.local.
mail.companyA.local A	MX 이름을 실제 IP로 바꿀 수 있는가	192.168.10.10
companyB.local MX	상대 zone 질문이 베타로 포워딩되는가	10 mail.companyB.local.
mail.companyB.local A	상대 서버 IP까지 찾을 수 있는가	192.168.20.10
-x 192.168.20.10	상대 역방향 질문도 포워딩되는가	mail.companyB.local.

14.2 클라이언트에서 서버를 직접 지정

Client A

```
dig @192.168.10.10 companyB.local MX +short
dig @192.168.10.10 mail.companyB.local A +short
```

Client B

```
dig @192.168.20.10 companyA.local MX +short
dig @192.168.20.10 mail.companyA.local A +short
```

DNS 단계의 완료 기준

양쪽 client가 자기 DNS 서버만 물어도 상대 도메인의 MX와 mail 호스트 A를 모두 얻는다. 이 단계가 실패하면 Postfix 설정으로 넘어가지 않는다.

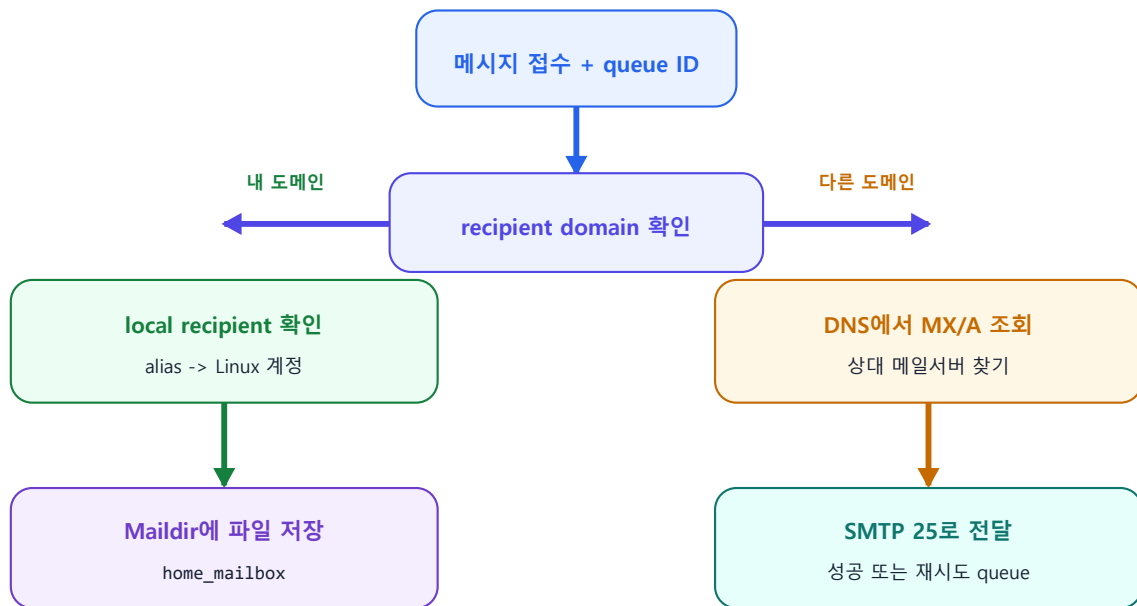
PART 5

Postfix가 접수하고 전달하는 과정

15. Postfix는 주소를 보고 두 갈래로 결정한다

Postfix가 메시지를 받으면 수취인 주소의 domain을 본다. 자기 domain이면 로컬 우편함에 저장하고, 다른 domain이면 DNS로 상대 MTA를 찾아 전달한다.

그림 15-1. Postfix의 핵심 결정 트리



mydestination은 '내가 최종 수신할 도메인'이고 mynetworks는 '나에게 외부 도메인 relay를 맡길 수 있는 신뢰 client 범위'다.

15.1 main.cf 핵심 이름을 먼저 이해하기

설정	쉬운 의미	비유
myhostname	이 Postfix 서버의 FQDN	우체국 지점 이름
mydomain	이 서버가 속한 메일 도메인	담당 지역
mydestination	최종 도착지로 받아들일 도메인	우리 지점이 직접 배달할 지역
mynetworks	외부 도메인 relay를 맡길 수 있는 신뢰 client	발송을 맡길 수 있는 고객/지점
home_mailbox	로컬 수신자의 Maildir 상대 경로	개인 우편함 위치
relay restrictions	누구의 어떤 목적지 전달을 허용할지 결정	무단 중계 방지 규칙

open relay를 막는 이유

누구나 알파에게 아무 도메인 메일이나 대신 보내 달라고 할 수 있으면 스팸 중계 서버가 된다. 그래서 실습 client subnet 만 mynetworks에 넣고, 목적지도 허가된 경우만 받는다.

16. 사용자 계정과 표시 주소 만들기

메일 주소는 userA/userB처럼 대문자를 포함하지만 Linux 계정은 usera/userb 소문자로 만든다. Postfix virtual alias가 표시 주소를 실제 계정에 연결한다.

16.1 설치와 계정

메일서버 알파

```
dnf install -y postfix
useradd --create-home --shell /sbin/nologin usera
passwd usera
install -d -m 0700 -o usera -g usera \
  /home/usera/Maildir/cur \
  /home/usera/Maildir/new \
  /home/usera/Maildir/tmp
restorecon -RFv /home/usera/Maildir
```

메일서버 베타

```
dnf install -y postfix
useradd --create-home --shell /sbin/nologin userb
passwd userb
install -d -m 0700 -o userb -g userb \
  /home/userb/Maildir/cur \
  /home/userb/Maildir/new \
  /home/userb/Maildir/tmp
restorecon -RFv /home/userb/Maildir
```

16.2 alias가 하는 일

무엇을 하는가	왜 필요한가	정상이라면
userA@companyA.local을 Linux 계정 usera에, userB@companyB.local을 userb에 연결한다.	표시용 이메일 주소와 운영체제 계정 이름의 대소문자/형식을 분리하기 위해서다.	postmap -q 결과가 각각 usera와 userb로 나온다.

17. 메일서버 알파의 Postfix 설정

17.1 /etc/postfix/main.cf

메일서버 알파 - 복사 가능한 기준 설정

```
compatibility_level = 2

myhostname = mail.companyA.local
mydomain = companyA.local
myorigin = $mydomain

inet_interfaces = all
inet_protocols = ipv4

mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
mynetworks = 127.0.0.0/8, 192.168.10.0/24

home_mailbox = Maildir/
virtual_alias_maps = hash:/etc/postfix/virtual

smtpd_relay_restrictions = permit_mynetworks, reject_unauth_destination
unknown_local_recipient_reject_code = 550
disable_vrfy_command = yes

recipient_delimiter = +
mailbox_size_limit = 0
message_size_limit = 10485760
```

17.2 /etc/postfix/virtual과 시작

메일서버 알파

```
userA@companyA.local    usera

# map    ,    ,
postmap /etc/postfix/virtual
postmap -q 'userA@companyA.local' hash:/etc/postfix/virtual
postfix check
systemctl enable --now postfix
postconf -n
ss -lntp | grep ':25 '
```

정상이라면

postmap -q가 usera를 출력하고, ss에 0.0.0.0:25 또는 서버 주소의 :25가 보인다. postfix check가 오류를 내지 않는다.

18. 메일서버 베타의 Postfix 설정

18.1 /etc/postfix/main.cf

메일서버 베타 - 복사 가능한 기준 설정

```
compatibility_level = 2

myhostname = mail.companyB.local
mydomain = companyB.local
myorigin = $mydomain

inet_interfaces = all
inet_protocols = ipv4

mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
mynetworks = 127.0.0.0/8, 192.168.20.0/24

home_mailbox = Maildir/
virtual_alias_maps = hash:/etc/postfix/virtual

smtpd_relay_restrictions = permit_mynetworks, reject_unauth_destination
unknown_local_recipient_reject_code = 550
disable_vrfy_command = yes

recipient_delimiter = +
mailbox_size_limit = 0
message_size_limit = 10485760
```

18.2 /etc/postfix/virtual과 시작

메일서버 베타

```
userB@companyB.local    userb

postmap /etc/postfix/virtual
postmap -q 'userB@companyB.local' hash:/etc/postfix/virtual
postfix check
systemctl enable --now postfix
postconf -n
ss -lntp | grep ':25 '
```

18.3 relay 허용과 로컬 수신을 구분하기

상황	알파의 판단	결과	관계 설정
Client A -> userB@companyB.local	외부 도메인, 신뢰 client	companyB.local로 relay	mynetworks
제3 client -> userB@companyB.local	외부 도메인, 비신뢰 client	Relay access denied	smtpd_relay_restrictions
Beta -> userB@companyB.local	자기 최종 도메인	userb Maildir에 로컬 배달	mydestination + alias

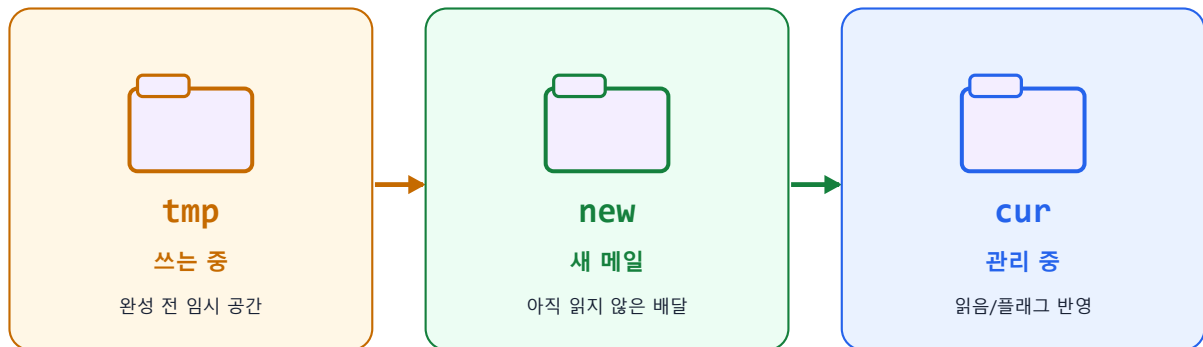
PART 6

Maildir 저장과 Dovecot 열람

19. Maildir는 세 개의 방으로 나뉜다

Postfix local(8)이 새 메일을 파일로 만들고, Dovecot이 같은 디렉터리를 읽는다. 두 프로그램이 같은 경로를 바라봐야 한다.

그림 19-1. Maildir의 tmp, new, cur



Maildir는 메일 1통을 파일 1개로 저장한다. 이 실습에서는 새 메일이 new에 나타난다.

실습에서 새 메일 파일은 먼저 /home/userb/Maildir/new에 나타난다. IMAP이 메시지를 관리하면 cur로 이동할 수 있다.

19.1 경로 일치가 중요한 이유

프로그램	경로를 정하는 설정	기준 값	역할
Postfix	home_mailbox	Maildir/	사용자 홈 아래 Maildir에 최종 저장
Dovecot	mail_location	maildir:~/Maildir	같은 Maildir를 IMAP/POP3로 제공

전형적인 증상

베타의 Maildir/new에는 파일이 있는데 IMAP에서 0 EXISTS라면 메일 전송이 아니라 Dovecot의 mail_location, 로그인 이름 정규화, 파일 권한을 먼저 확인한다.

20. Dovecot은 우편함을 열어 주는 창구다

그림 20-1. 사용자 B가 Maildir를 직접 읽지 않는 이유



사용자 B는 서버 파일을 직접 열지 않는다. Dovecot에 로그인하고, Dovecot이 권한을 확인한 뒤 목록과 본문을 프로토콜 응답으로 제공한다.

20.1 설치와 공통 설정

두 서버

```
dnf install -y dovecot
dovecot --version
```

/etc/dovecot/dovecot.conf

```
protocols = imap pop3
listen = *

!include conf.d/*.conf
!include_try local.conf
```

/etc/dovecot/conf.d/10-mail.conf

```
mail_location = maildir:~/Maildir
first_valid_uid = 1000

namespace inbox {
  inbox = yes
}
```

20.2 인증 이름과 평문 실습 설정

사용자는 userB@companyB.local로 로그인하지만 Linux 계정은 userb다. %Ln은 domain을 제외한 local-part를 소문자로 정규화한다. 이 실습은 서버 하나가 도메인 하나만 담당하므로 단순하게 사용할 수 있다.

/etc/dovecot/conf.d/10-auth.conf

```
disable_plaintext_auth = no
auth_mechanisms = plain login
auth_username_format = %Ln

!include auth-system.conf.ext
```

/etc/dovecot/conf.d/10-ssl.conf

```
ssl = no
```

평문 인증 경고

비밀번호와 메일 내용을 눈으로 관찰하기 위한 폐쇄망 전용 설정이다. 인터넷 연결망이나 실제 사용자 환경에서는 사용하지 않는다. 운영 확장은 TLS 절에서 다룬다.

20.3 검사와 시작

메일서버 알파

```
doveconf -n
systemctl enable --now dovecot
doveadm auth test userA@companyA.local
ss -lntp | grep -E ':(110|143) '
```

메일서버 베타

```
doveconf -n
systemctl enable --now dovecot
doveadm auth test userB@companyB.local
ss -lntp | grep -E ':(110|143) '
```

검사	확인하는 것	정상 예
doveconf -n	실제로 적용된 설정	mail_location과 protocols가 보임
doveadm auth test	주소 형식이 Linux 계정으로 인증되는가	passdb: userb auth succeeded
ss -lntp	IMAP/POP3 창구가 열렸는가	:143, :110 LISTEN

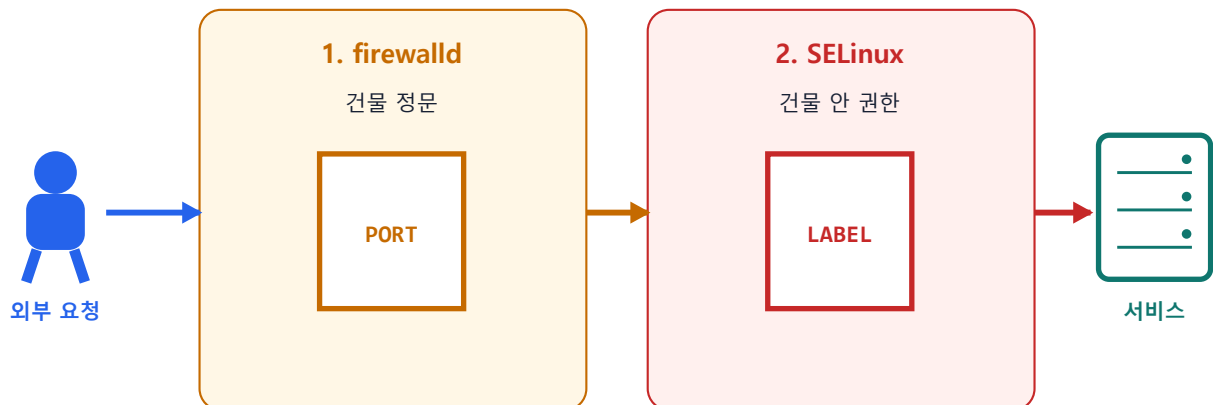
PART 7

접근 허용과 파일 권한을 따로 보기

21. firewalld와 SELinux는 서로 다른 문이다

firewalld는 네트워크에서 어느 서비스 창구로 들어올 수 있는지 정한다. SELinux는 서버 안에서 프로세스가 어떤 파일과 기능에 접근할 수 있는지 정한다.

그림 21-1. 건물 정문과 건물 안 권한



포트가 열려도 SELinux 권한이 맞지 않으면 파일 접근은 실패할 수 있다.

정문을 통과한 뒤에도 내부 권한이 맞아야 작업이 성공한다. 두 보안 계층을 서로 대신하는 설정으로 생각하지 않는다.

21.1 firewalld

두 서버

```

systemctl enable --now firewalld
firewall-cmd --get-active-zones

firewall-cmd --permanent --add-port=53/udp
firewall-cmd --permanent --add-port=53/tcp
firewall-cmd --permanent --add-port=25/tcp
firewall-cmd --permanent --add-port=143/tcp
firewall-cmd --permanent --add-port=110/tcp
firewall-cmd --reload
firewall-cmd --list-ports
  
```

포트	그림의 어느 통로인가
53/udp, 53/tcp	DNS의 일반 질문과 큰 응답/재시도
25/tcp	Client의 기준 실습 제출과 MTA 간 전달
143/tcp	IMAP 평문 실습
110/tcp	POP3 평문 실습

21.2 SELinux

Maildir가 사용자 홈 아래 있으므로 Dovecot의 홈 디렉터리 접근 boolean을 확인한다. Rocky Linux point release와 정책 패키지에 따라 boolean 유무가 다를 수 있어 먼저 존재 여부를 검사한다.

두 서버

```
getenforce
getsebool -a | grep -E 'dovecot|postfix|named'

if getsebool dovecot_enable_homedirs >/dev/null 2>&1; then
    setsebool -P dovecot_enable_homedirs on
fi

restorecon -RFv /home/usera/Maildir #
restorecon -RFv /home/userb/Maildir #
```

정적 BIND primary zone은 파일을 읽기만 하므로 named_write_master_zones를 무조건 켤 필요가 없다. dynamic update처럼 named가 zone 파일을 직접 써야 할 때만 검토한다.

거부 기록 확인

```
ausearch -m AVC,USER_AVC -ts recent | audit2why
ls -ldZ /home/usera/Maildir 2>/dev/null
ls -ldZ /home/userb/Maildir 2>/dev/null
```

SELinux를 끄는 대신 원인을 찾기

소유권, 일반 권한, file labeling을 먼저 확인한다. 원인을 모른 채 audit2allow로 정책을 만들면 필요 이상 권한을 영구 허용할 수 있다.

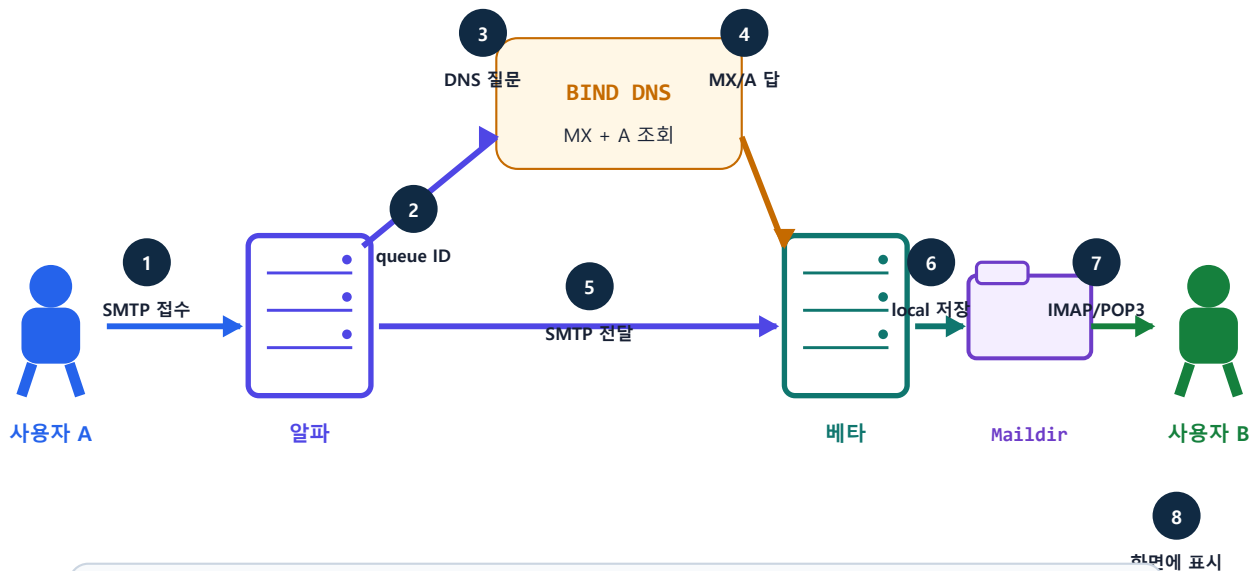
PART 8

사용자 A에서 사용자 B까지 실제로 따라가기

22. 한 통의 메일을 8개 번호로 추적한다

실습의 목적은 단지 '메일이 왔다'를 보는 것이 아니다. 각 번호에서 어떤 프로그램이 결정하고 무엇을 증거로 남기는지 연결하는 것이다.

그림 22-1. Client A -> 알파 -> 베타 -> Client B 전체 경로



한 통을 따라갈 때는 1 -> 8 순서로 확인한다.
어느 번호에서 멈췄는지 알면 고칠 설정 범위가 작아진다.

번호 1-5는 발신과 서버 간 전달, 6은 실제 저장, 7-8은 사용자 열람이다. DNS는 3-4에서 목적지를 안내한다.

22.1 시작 전 6개 질문

- Client A에서 mail.companyA.local이 192.168.10.10으로 해석되는가?
- 알파에서 companyB.local MX와 mail.companyB.local A가 정상인가?
- Postfix 알파/베타가 25번 포트에 대기 중인가?
- Dovecot 베타가 143/110번 포트에 대기 중인가?
- userb의 Maildir/new가 있고 소유자가 userb인가?
- userB@companyB.local의 비밀번호를 알고 있는가?

실습 경로를 한 문장으로 말하기

Client A가 SMTP로 알파에 제출하고, 알파가 DNS로 베타를 찾아 SMTP로 넘기며, 베타 Postfix가 userb Maildir에 저장하고, Client B가 Dovecot의 IMAP 또는 POP3로 읽는다.

23. swaks로 전체 발신 경로를 한 번에 검증

--server는 상대 베타가 아니라 자기 발신 서버 알파다. Client A가 알파에 맡기고 알파가 MX를 따라 베타로 전달하는 흐름을 보기 위해서다.

Client A

```
swaks \
--server mail.companyA.local \
--port 25 \
--ehlo clientA.companyA.local \
--from userA@companyA.local \
--to userB@companyB.local \
--header 'Subject: A-to-B visual beginner lab' \
--header 'From: User A <userA@companyA.local>' \
--header 'To: User B <userB@companyB.local>' \
--body 'Hello through Postfix Alpha and Postfix Beta.'
```

23.1 출력에서 순서대로 찾을 것

보이는 조각	그림 22-1의 번호	뜻
220 ... ESMTP Postfix	1	알파의 SMTP 창구가 열려 있음
RCPT TO 뒤 250	1	알파가 B 도메인 relay 요청을 수락
250 ... queued as ID	2	알파가 메시지 책임을 인수
relay=mail.companyB.local[192.168.20.10]:25	3-5	DNS 결과를 사용해 베타에 전달
status=sent	5	베타가 메시지를 수락
delivered to maildir	6	베타가 userb Maildir에 저장

250은 목적지 화면에 보였다는 뜻이 아니다

알파의 250 queued는 알파가 책임을 인수했다는 뜻이다. 그 뒤 알파->베타 전달과 베타의 local 저장이 성공해야 사용자 B가 읽을 수 있다.

24. nc로 SMTP 명령을 직접 말해 보기

swaks는 자동으로 대화를 만들어 준다. nc 실습은 어떤 줄이 envelope이고 어떤 줄이 header/body이며 어디서 입력이 끝나는지 손으로 확인한다.

Client A

```
nc mail.companyA.local 25

EHLO clientA.companyA.local
MAIL FROM:<userA@companyA.local>
RCPT TO:<userB@companyB.local>
DATA
From: User A <userA@companyA.local>
To: User B <userB@companyB.local>
Subject: Manual SMTP lab

This is the message body.
.
QUIT
```

24.1 빈 줄과 단독 점은 서로 다르다

입력	의미	빠뜨리면
Subject 다음 빈 줄	header와 body의 경계	본문이 header처럼 잘못 해석될 수 있음
새 줄의 단독 점	SMTP DATA 전체 입력 종료	Postfix가 계속 다음 입력을 기다림
QUIT	SMTP 대화 정상 종료	연결을 닫으면 끝나지만 종료 응답을 못 받음

telnet을 사용할 때

telnet mail.companyA.local 25로도 같은 명령을 입력할 수 있다. 폐쇄망 평문 실습에서만 사용하고, 운영 환경에서는 인증과 TLS를 갖춘 메일 client를 사용한다.

25. 베타의 디스크에서 저장 사실 확인

이 단계는 '전송'과 '열람' 사이의 경계다. Maildir/new에 파일이 있으면 DNS, 서버 간 SMTP, 베타의 Postfix local delivery까지는 성공했다.

메일서버 베타

```
find /home/userb/Maildir -maxdepth 2 -type f -printf '%p\n'
ls -la /home/userb/Maildir/new
sed -n '1,60p' /home/userb/Maildir/new/*
```

무엇을 하는가	왜 필요한가	정상이라면
userb Maildir의 실제 파일과 header/body를 확인한다.	IMAP 문제와 메일 전송 문제를 분리한다. 파일이 있으면 메시지는 이미 사용자 사서함까지 도착했다.	new 아래 파일 1개 이상과 Subject: A-to-B visual beginner lab이 보인다.

25.1 파일이 없을 때 볼 것

- 베타 Postfix 로그에 user unknown 또는 status=bounced가 있는지 본다.
- postmap -q userB@companyB.local hash:/etc/postfix/virtual이 userb를 출력하는지 본다.
- getent passwd userb와 Maildir 소유권을 확인한다.
- 알파 queue에 status=deferred로 남아 있으면 베타 도착 전 문제다.

26. Client B에서 IMAP으로 우편함 열기

이제 저장된 파일을 직접 읽지 않고 사용자 B의 관점으로 Dovecot에 로그인한다. tag가 붙은 명령마다 같은 tag의 OK 응답이 돌아오는지 본다.

Client B

```
nc mail.companyB.local 143

a001 CAPABILITY
a002 LOGIN userB@companyB.local "<PASSWORD>"
a003 SELECT INBOX
a004 FETCH 1 (FLAGS BODY.PEEK[HEADER.FIELDS (FROM TO SUBJECT DATE)])
a005 FETCH 1 BODY.PEEK[]
a006 LOGOUT
```

응답	그림 22-1에서 확인하는 것	쉬운 의미
a002 OK	7	사용자 인증 성공
* 1 EXISTS	7	선택한 INBOX에 메시지 1개
FETCH ... Subject	8	메일 header를 실제로 읽음
BODY.PEEK[]	8	읽음 상태를 불필요하게 바꾸지 않고 전체 본문 조회

26.1 Maildir에는 있지만 0 EXISTS라면

1. `doveadm auth test userB@companyB.local`로 로그인 정규화를 확인한다.
2. `doveconf -n | grep -E "mail_location|auth_username_format"`로 실제 적용값을 본다.
3. Maildir와 하위 디렉터리 소유자가 userb인지 확인한다.
4. SELinux AVC 기록이 있는지 확인하고 `restorecon`을 수행한다.

27. POP3로 같은 메시지를 확인하고 반대 방향도 시험

27.1 Client B의 POP3 확인

Client B

```
nc mail.companyB.local 110

USER userB@companyB.local
PASS <PASSWORD>
STAT
LIST
RETR 1
QUIT
```

평문에서 직접 확인한 교육 포인트

패킷의 애플리케이션 내용을 보면 LOGIN/PASS와 메시지 본문이 그대로 보인다. 이 사실을 확인한 뒤 TLS 확장으로 넘어가면 암호화가 왜 필요한지 자연스럽게 이해할 수 있다.

27.2 사용자 B -> 사용자 A 반대 방향

Client B

```
swaks \
--server mail.companyB.local \
--port 25 \
--ehlo clientB.companyB.local \
--from userB@companyB.local \
--to userA@companyA.local \
--header 'Subject: B-to-A reverse test' \
--body 'Reverse direction works.'
```

무엇을 하는가	왜 필요한가	정상이라면
베타를 발신 MTA로, 알파를 수신 MTA로 사용해 반대 방향을 보낸다.	조건부 DNS 포워딩, MX, 방화벽, Postfix 설정이 한쪽에만 맞아 있는 실수를 찾는다.	알파의 /home/userA/Maildir/new에 메시지 파일이 생기고 IMAP으로 Subject를 읽는다.

PART 9

로그와 패킷에서 같은 흐름 찾기

28. queue ID는 메시지의 배송 조회 번호다

Postfix는 메시지를 접수할 때 queue ID를 발급한다. 알파와 베타는 서로 다른 queue ID를 쓸 수 있으므로 알파 로그의 queued as 문구로 두 번호를 연결한다.

그림 28-1. 알파 ID와 베타 ID 연결



queue ID는 택배 운송장 번호와 같다. 서버가 바뀌면 새 번호가 생길 수 있다.

한 서버 안에서는 같은 queue ID를 따라가고, 서버가 바뀌는 지점에서는 queued as로 새 ID를 찾는다.

28.1 로그와 queue 명령

메일서버 알파/베타

```

journalctl -u postfix -f
journalctl -u dovecot -f
journalctl -u named -f
tail -F /var/log/maillog

postqueue -p
postcat -q <QUEUE_ID>
postqueue -f
  
```

Rocky Linux 9에서는 systemd journal을 기본 증거로 사용할 수 있다. /var/log/maillog는 rsyslog와 해당 규칙이 활성화된 경우에 존재한다.

명령	무엇을 보는가
postqueue -p	대기/재시도 중인 메시지와 이유
postcat -q ID	queue에 저장된 envelope와 message
postqueue -f	DNS/네트워크 수정 뒤 deferred queue 즉시 재시도
journalctl -u dovecot	인증, Maildir 접근, IMAP/POP3 오류
journalctl -u named	zone load, query, forward 관련 오류

28.2 Postfix 로그 한 줄 번역

예시

```
postfix/smtp[2201]: 9A1BC12345:
to=<userB@companyB.local>,
relay=mail.companyB.local[192.168.20.10]:25,
dsn=2.0.0,
status=sent (250 2.0.0 Ok: queued as 7F33A45678)
```

로그 조각	뜻	그림의 위치
9A1BC12345	알파가 발급한 queue ID	2
to=<...>	실제 SMTP envelope recipient	1-5
relay=...:25	실제로 전달한 베타 주소와 포트	5
dsn=2.0.0	성공 계열 전달 상태	5
queued as 7F33A45678	베타가 발급한 새 queue ID	5-6

28.3 애플리케이션 대화 캡처

SMTP, DNS, IMAP, POP3 저장

```
tcpdump -ni any -s0 -w /tmp/mail-lab.pcap \
'(port 25 or port 53 or port 110 or port 143)'

# Wireshark display filters
dns
smtp
imap
pop
```

패킷에서는 TCP 연결 세부보다 Wireshark의 Follow TCP Stream을 이용해 SMTP/IMAP/POP3 명령 순서와 평문 내용을 확인한다.

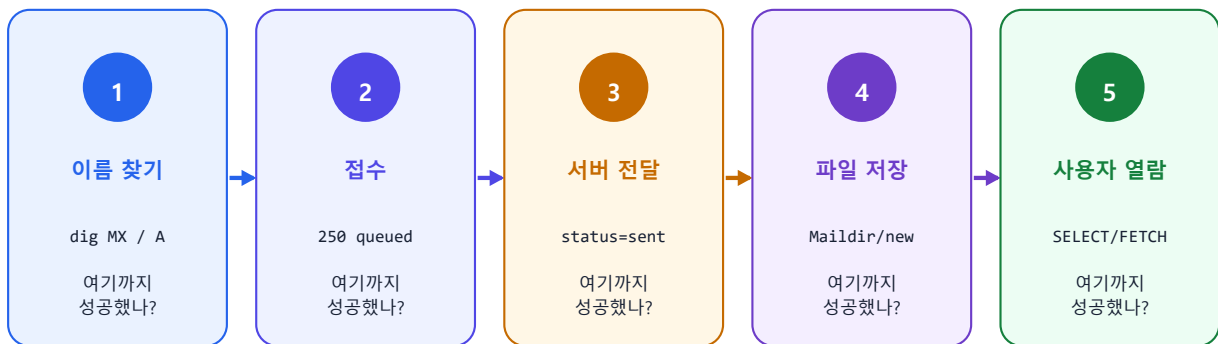
PART 10

처음 성공하지 않을 때 단계별로 좁히기

29. 증상보다 먼저 멈춘 단계를 찾는다

'메일이 안 된다'는 말만으로는 범위가 너무 넓다. DNS, 접속, 서버 간 전달, 파일 저장, 사용자 열람을 한 칸씩 확인한다.

그림 29-1. 앞 단계가 성공해야 다음 단계로 간다



앞 단계가 성공해야 다음 단계를 조사한다.

여러 설정을 한꺼번에 바꾸면 원인과 해결책을 연결할 수 없다.

오른쪽 끝에서 문제가 보여도 실제 원인은 왼쪽 단계일 수 있다. 항상 1번부터 증거를 확인한다.

29.1 초심자용 증상 표

증상	쉬운 해석	첫 확인
dig timeout	DNS 서버와 대화 자체가 안 됨	route, firewall 53, named listen
SERVFAIL	DNS가 내부 문제로 답을 못 만들	named-checkconf, zone load log
NXDOMAIN	그 DNS 이름이 없다고 판단	record 이름, zone, trailing dot
Host or domain not found	Postfix가 상대 MX/A를 못 찾음	서버에서 dig MX/A
Relay access denied	client가 외부 목적지 relay 권한 없음	mynetworks, mydestination
User unknown	주소를 실제 local user로 연결 못 함	getent passwd, postmap -q
status=deferred	지금은 전달하지 못해 queue에서 재시도	postqueue -p의 reason
Maildir에는 있으나 0 EXISTS	Dovecot이 다른 위치/사용자를 보고 있음	mail_location, 로그인 이름
Authentication failed	계정 정규화 또는 PAM/비밀번호 실패	doveadm auth test
Permission denied / AVC	일반 권한 또는 SELinux 차단	ls -lZ, restorecon, ausearch

30. 고의로 한 곳만 망가뜨려 배우기

정상 구성을 백업한 뒤 한 번에 하나의 오류만 만든다. 예상한 단계에서 정확히 실패하는지 확인하고 곧바로 복구한다.

고의 오류	예상 결과	배울 개념	복구 후 증거
MX target의 끝 점 제거	이름이 zone 뒤에 다시 붙음	FQDN과 상대 이름	dig MX/A 정상
remote forwarder IP 오타	상대 MX timeout/SERVFAIL	조건부 포워딩	상대 MX 응답
mynetworks에서 client subnet 제거	Relay access denied	relay 권한	RCPT TO 뒤 250
mydestination에서 \$mydomain 제거	자기 도메인을 최종 목적지로 받지 않음	local destination	베타 local delivery
Dovecot mail_location을 다른 경로로 변경	파일은 있지만 IMAP 0 EXISTS	저장 경로 일치	SELECT에서 EXISTS

30.1 복구 원칙

- 오류를 만들기 전 원본 설정 파일을 별도 이름으로 복사한다.
- 증상이 나타나면 그림 29-1의 어느 번호까지 성공했는지 기록한다.
- 오류를 원복하고 검사 명령을 먼저 실행한 뒤 서비스를 reload/restart한다.
- 같은 검증 명령으로 정상 복구 증거를 남긴 뒤 다음 오류로 넘어간다.

한꺼번에 여러 설정을 바꾸지 말 것

여러 파일을 동시에 바꾸면 무엇이 문제였고 무엇이 해결했는지 학습할 수 없다. 한 단계의 원인과 증거를 연결하는 것이 실습의 핵심이다.

PART 11

기존 실습을 운영 구조로 확장하기

31. Dovecot LMTP와 TLS 확장

31.1 Dovecot을 실제 MDA로 만들기

배달 담당자를 바꾸는 확장

기존 구성에서는 Postfix local(8)이 Maildir에 넣는다. 아래 구성은 Postfix가 Dovecot LMTP에 마지막 배달을 맡기도록 바꾼다. home_mailbox = Maildir/와 동시에 사용하지 않는다.

Dovecot protocol과 LMTP socket

```
# /etc/dovecot/dovecot.conf
protocols = imap pop3 lmtp

# /etc/dovecot/conf.d/10-master.conf
service lmtp {
    unix_listener /var/spool/postfix/private/dovecot-lmtp {
        mode = 0600
        user = postfix
        group = postfix
    }
}
```

Postfix main.cf 변경

```
#
# home_mailbox = Maildir/

mailbox_transport = lmtp:unix:private/dovecot-lmtp
```

검사

```
doveconf -n
postfix check
systemctl restart dovecot postfix
ls -l /var/spool/postfix/private/dovecot-lmtp
```

31.2 TLS는 무엇을 바꾸는가

TLS는 SMTP/IMAP/POP3 연결 구간에서 명령, 비밀번호, 메시지 내용을 보호한다. Maildir 파일 자체가 서버 디스크에서 자동으로 암호화되는 것은 아니다.

구간	기준 실습	운영 권장
사용자 발신	신뢰 subnet의 SMTP 25	587 STARTTLS 또는 465 TLS + SMTP AUTH
서버 간 전달	SMTP 25 평문	SMTP 25에서 정책에 맞는 TLS
메일 읽기	IMAP 143 / POP3 110 평문	IMAPS 993 / POP3S 995

Dovecot TLS 개념 예

```
ssl = required
ssl_min_protocol = TLSv1.2
ssl_cert = </etc/pki/tls/certs/mail.crt
ssl_key = </etc/pki/tls/private/mail.key
```

포트만 열면 끝나지 않는다

465/587은 방화벽 포트를 여는 것만으로 완성되지 않는다. Postfix master.cf 서비스, SMTP AUTH, Dovecot SASL socket, 인증서, relay 정책을 함께 구성해야 한다.

32. 강의 진행 순서와 완료 체크리스트

32.1 설명 순서

1. 그림 1-1로 기술 용어 없이 다섯 장면을 학습자가 자기 말로 설명하게 한다.
2. 그림 2-1에서 이미 이해한 역할에 SMTP, DNS, Postfix, Maildir, Dovecot, IMAP 이름표를 붙인다.
3. 그림 6-1과 7-1로 보내기와 읽기 프로토콜을 구분한다.
4. 그림 9-1에 실제 GNS3 IP를 표시하고 양방향 라우팅부터 확인한다.
5. DNS -> Postfix -> Maildir -> Dovecot 순서로 각 단계를 완료할 때마다 증거를 남긴다.
6. 마지막에는 그림 22-1의 1-8 번호를 queue ID와 로그에 연결한다.

32.2 설명 체크

- [] 수강생이 MX와 A의 차이를 '담당 서버 이름'과 '실제 IP'로 설명한다.
- [] SMTP는 보내는 길, IMAP/POP3는 저장된 메일을 읽는 길이라고 설명한다.
- [] mydestination과 mynetworks의 차이를 자기 도메인과 relay 권한으로 설명한다.
- [] 기준 구성에서 Postfix local(8)과 Dovecot의 역할을 정확히 구분한다.
- [] envelope와 message header를 구분한다.
- [] Postman과 Postfix가 전혀 다른 도구임을 설명한다.

32.3 구축 체크

- [] 양쪽 DNS에서 상대 MX/A/PTR이 정상이다.
- [] Client A -> 알파에서 250 queued를 받는다.
- [] 알파 로그에 베타 IP:25와 status=sent가 있다.
- [] 베타의 userb Maildir/new에 메시지 파일이 있다.
- [] IMAP SELECT에서 EXISTS가 보이고 FETCH로 Subject를 읽는다.
- [] B -> A 반대 방향도 같은 증거로 성공한다.
- [] 제3 client의 임의 relay 요청은 거부된다.

33. 한 장 용어 요약

용어	가장 짧은 설명	그림에서의 위치
SMTP	새 메일을 client->서버, 서버->서버로 보냄	사용자 A -> 알파 -> 베타
POP3	저장된 메시지 목록과 내용을 단순 다운로드	베타/Dovecot -> 사용자 B
IMAP	서버 우편함의 폴더와 상태를 원격 관리	베타/Dovecot <-> 사용자 B
MX	도메인의 메일 담당 서버 이름	알파가 베타를 찾는 DNS 질문
A	서버 이름의 IPv4 주소	mail.companyB.local -> 192.168.20.10
MTA	메일을 queue에 두고 다른 서버로 전달	Postfix
MDA/LDA	최종 사서함에 저장	기준: Postfix local(8)
MRA	IMAP/POP3로 저장 메일 제공	Dovecot
queue ID	Postfix 메시지 추적 번호	250 queued와 로그
relay	다른 도메인/서버로 메시지를 넘김	알파 -> 베타
Maildir	메일 1통을 파일 1개로 저장	userb의 개인 우편함
envelope	SMTP가 실제 배달에 쓰는 sender/recipient	MAIL FROM/RCPT TO
header	사용자 화면에 표시되는 From/To/Subject	DATA 안의 표시 정보

마지막 자가 설명

아무 약어 없이 한 통의 여행을 먼저 설명한 뒤, 각 장면에 기술 이름을 다시 붙일 수 있으면 이 교재의 개념 목표를 달성한 것이다.

34. 공식 참고 자료

프로토콜 의미와 제품 설정은 아래 공식 문서를 기준으로 확인했다. Rocky Linux 9 point release와 package update에 따라 기본값이 달라질 수 있으므로 실제 시스템의 man page와 검사 명령을 함께 사용한다.

1. [RFC 5321 - Simple Mail Transfer Protocol](https://www.rfc-editor.org/rfc/rfc5321.html)

<https://www.rfc-editor.org/rfc/rfc5321.html>

2. [RFC 1939 - Post Office Protocol Version 3](https://www.rfc-editor.org/rfc/rfc1939.html)

<https://www.rfc-editor.org/rfc/rfc1939.html>

3. [RFC 9051 - Internet Message Access Protocol Version 4rev2](https://www.rfc-editor.org/rfc/rfc9051.html)

<https://www.rfc-editor.org/rfc/rfc9051.html>

4. [RFC 8314 - TLS for Email Submission and Access](https://www.rfc-editor.org/rfc/rfc8314.html)

<https://www.rfc-editor.org/rfc/rfc8314.html>

5. [RFC 5598 - Internet Mail Architecture](https://www.rfc-editor.org/rfc/rfc5598.html)

<https://www.rfc-editor.org/rfc/rfc5598.html>

6. [ISC BIND 9 Administrator Reference Manual](https://bind9.readthedocs.io/en/stable/)

<https://bind9.readthedocs.io/en/stable/>

7. [Postfix Basic Configuration](https://www.postfix.org/BASIC_CONFIGURATION_README.html)

https://www.postfix.org/BASIC_CONFIGURATION_README.html

8. [Dovecot 2.3 Documentation](https://doc.dovecot.org/2.3/)

<https://doc.dovecot.org/2.3/>

9. [RHEL 9 - Configuring a BIND DNS server](https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html/managing_networking_infrastructure_services/assembly_setting-up-and-configuring-a-bind-dns-server_networking-infrastructure-services)

https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html/managing_networking_infrastructure_services/assembly_setting-up-and-configuring-a-bind-dns-server_networking-infrastructure-services

10. [RHEL 9 - Deploying mail servers](https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html-single/deploying_mail_servers/index)

https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html-single/deploying_mail_servers/index

이번 개정판의 범위

기술 용어가 나오기 전에 사용자 A, 메일서버 알파, 메일서버 베타, 사용자 B의 흐름을 여러 그림으로 설명했다. 같은 그림에 기술 이름을 단계적으로 붙이고, 각 설정과 검증 명령을 그 흐름의 번호에 연결했다.

END OF VISUAL BEGINNER MAIL LAB MANUAL