SRX 演習（SRX-1 / SRX-2）解説ノート（例題コマンド＋コマンド解説付き）

> 目的：演習文の「何をやればいいか」を、**手順**と**理由**に分解して整理。
> さらに、各コマンドを **(1) 書式 → (2) 何をする？ → (3) 例題（サンプル値入り） → (4) 確認** の形で統一してまとめています。
>
> <> はあなたの環境（IF名、IP、ゾーン名）に合わせて置き換えてください。

---

0. 演習文（原文）

■SRX-1

演習人数：4〜5人
用意するもの：チームで2台 EC2 インスタンス（Webサーバー）

新たに Web サーバーをホストすることになりました。新たに1台 Web サーバーを構築し、外部から Web サーバーへのアクセスを許可する設定を追加してみましょう。
（1人1回は CLI で設定できるよう何周か回してください。）

■SRX-2

Web サーバーへのアクセスは HTTP のみ許可とし、それ以外の通信は拒否するようにしてください。
また、不正なアクセスを検知しブロックする設定を入れてください。

---

1. 全体像（この順で組むと詰まりにくい）

1. **ゾーン作成**（untrust / trust）
2. **Destination NAT**（外部IP → Webサーバ内部IP）
3. **セキュリティポリシー**（通す/止める）
4. **screen（IDS/DoS検知）**（不正検知・ブロック）
5. **確認コマンド**（hit-count / NATルール / screen統計）

---

2. 前提モデル（これを想像すると理解が速い）

- `untrust`：インターネット側（外部IF）
- `trust`：内部（Webサーバがいる側IF）
- 外部からの HTTP(80) を SRX で受けて **WebサーバのプライベートIPへ DNAT** で転送
- その通信を **untrust→trust のポリシーで許可**（SRX-2はHTTPのみ）
- さらに **screen** を untrust に適用して SYN flood 等を検知・抑止

> ⚠重要：DNAT の `match destination-address` は「外部から叩く宛先IP」です。
> AWS だと **EIP(パブリックIP)** の場合もあるし、演習構成によっては SRX の **untrust 側プライベートIP** の場合もあります。
> 「ブラウザ/ curl でアクセスする IP」がそのまま match に入る、と覚えると迷いません。

3. SRX-1（外部から Web サーバーへアクセス許可）

3.1 ゾーン作成（インターフェースを所属させる）

コマンド（書式）

```
set security zones security-zone <ゾーン名> interfaces <インターフェース名>
```

何をする？

- インターフェースを **どのセキュリティゾーンに属させるか**を定義します。
- ゾーンはポリシー（from/to）や NAT 適用条件の“土台”になります。

例題（サンプル）

```
set security zones security-zone untrust interfaces ge-0/0/0.0
set security zones security-zone trust   interfaces ge-0/0/1.0
```

確認

```
show security zones
show interfaces terse | match ge-0/0/0|ge-0/0/1
```

3.2 Destination NAT（外部IP → Webサーバ内部IP）

> ここが「外部公開」の本体（いわゆるポートフォワード/宛先変換）です。

3.2.1 変換先 pool を作る

コマンド（書式）

```
set security nat destination pool <pool名> address <内部IP/プレフィックス>
```

何をする？

- 「宛先をどこへ変換するか（＝Webサーバ内部IP）」を pool として定義します。

例題

```
set security nat destination pool pool_web address 172.31.1.48/32
```

確認

```
show security nat destination pool
```

3.2.2 どこから入ってきた通信に NAT を適用するか（from）

コマンド（書式）

```
set security nat destination rule-set <番号> from interface <インターフェース名>
```

何をする？

- 「どの入口（外部IFなど）から入った通信に、このDNATを適用するか」を決めます。

例題

```
set security nat destination rule-set 1 from interface ge-0/0/0.0
```

3.2.3 どの宛先IPを変換するか（match）

コマンド（書式）

```
set security nat destination rule-set <番号> rule <ルール名> match destination-address <外部から見える宛先IP/32>
```

何をする？

- 外から見える「宛先IP（ブラウザで叩くIP）」に来た通信だけを変換対象にします。

例題（EIPを叩く想定）

```
set security nat destination rule-set 1 rule web match destination-address 52.26.109.29/32
```

3.2.4 変換先 pool を紐付け（then）

コマンド（書式）

```
set security nat destination rule-set <番号> rule <ルール名> then
destination-nat pool <pool名>
```

何をする？

- match に合致した通信を、指定 pool（Web内部IP）へ DNAT します。

例題

```
set security nat destination rule-set 1 rule web then destination-nat pool
pool_web
```

3.2.5 DNAT 設定の確認

コマンド

```
show security nat destination rule all
show security nat destination pool
```

何を見る？

- rule-set / rule の **match と then が意図通りか**
- pool の **宛先IPが正しいか**

3.3 セキュリティポリシー（SRX-1：まずは通る状態を作る）

コマンド（書式）

```
set security policies from-zone <送信元ゾーン> to-zone <宛先ゾーン> policy <名
前> match source-address <src>
set security policies from-zone <送信元ゾーン> to-zone <宛先ゾーン> policy <名
前> match destination-address <dst>
set security policies from-zone <送信元ゾーン> to-zone <宛先ゾーン> policy <名
前> match application <app>
set security policies from-zone <送信元ゾーン> to-zone <宛先ゾーン> policy <名
前> then <permit|deny>
```

何をする？

- NAT で宛先が変換されても、SRX は **ポリシーが permit しない限り通しません。**
- SRX-1 は「外→内が通る状態」を作るのが狙いなので、まずは広めに permit して疎通を作るのが定石です。

例題（SRX-1：とりあえず any を許可）

```
set security policies from-zone untrust to-zone trust policy allow-web
match source-address any
set security policies from-zone untrust to-zone trust policy allow-web
match destination-address any
set security policies from-zone untrust to-zone trust policy allow-web
match application any
set security policies from-zone untrust to-zone trust policy allow-web
then permit
```

確認（SRX演習で最重要）

```
show security policies from-zone untrust to-zone trust
show security policies hit-count
```

> `hit-count` が増える＝そのルールに当たっている、です。
> 「通らない時の犯人探し」最短ルートになります。

3.4 AWS などで必要になることがある：proxy-arp（構成次第）

コマンド（書式）

```
set security nat proxy-arp interface <外部IF> address <外部IP>
```

何をする？

- SRX が指定IPに対して **ARP応答** できるようにします（構成によって必要）。
- 「外からIP宛に来てるのに、L2で届かない」系の詰まりを潰します。

例題

```
set security nat proxy-arp interface ge-0/0/0.0 address 52.26.109.29
```

確認

```
show security nat proxy-arp
```

---

4. SRX-2（HTTPのみ許可＋それ以外拒否＋不正検知/ブロック）

4.1 外→内ポリシーを HTTP のみに絞る

何をする？

- SRX-1 の「any permit」をやめて、HTTP(80) だけ許可します。
- それ以外は拒否（暗黙deny or 明示deny）。

4.1.1 まず SRX-1 の広い許可ルールを消す（例）

コマンド

```
delete security policies from-zone untrust to-zone trust policy allow-web
```

何をする？

- SRX-1 で作った「全部許可」を撤去し、次の HTTP only ルールに切り替えます。

4.1.2 HTTP only の permit ルールを作る

コマンド（例題）

```
set security policies from-zone untrust to-zone trust policy allow-http
match source-address any
set security policies from-zone untrust to-zone trust policy allow-http
match destination-address any
set security policies from-zone untrust to-zone trust policy allow-http
match application junos-http
set security policies from-zone untrust to-zone trust policy allow-http
then permit
```

何をする？

- `junos-http` を使って「HTTPアプリケーションだけ」許可します。
- これで 80 以外のアプリは基本的に通りません（※既存ルールがなければ）。

4.1.3 それ以外を拒否（明示 deny を入れるパターン）

> 演習で「拒否している」ことを分かりやすくするなら明示 deny が便利です（ルール順は **permit が先**）。

コマンド（例題）

```
set security policies from-zone untrust to-zone trust policy deny-rest
match source-address any
set security policies from-zone untrust to-zone trust policy deny-rest
match destination-address any
set security policies from-zone untrust to-zone trust policy deny-rest
match application any
set security policies from-zone untrust to-zone trust policy deny-rest
then deny
```

4.1.4 ポリシー確認（必ず hit-count を見る）

```
show security policies from-zone untrust to-zone trust
show security policies hit-count
```

4.2 screen（不正アクセス検知・ブロック）

> 演習資料の「IDS」は、ここでは主に **security screen**（DoS/異常パケット検知・抑止）を指します。
> `untrust` に適用するのが基本です。

4.2.1 IDS option（検知項目）を作る

コマンド（書式）

```
set security screen ids-option <スクリーン名> <プロトコル> <スクリーン種別> [パラメータ...]
```

何をする？

- どの攻撃/異常を、どの閾値で検知・抑止するかを定義します。
- 同じ スクリーン名 に対して複数行追加して"盛っていく"イメージです。

例題（演習資料の表に近い例）

```
set security screen ids-option screen_web icmp ping-death
set security screen ids-option screen_web ip source-route-option
```

```
set security screen ids-option screen_web ip tear-drop

set security screen ids-option screen_web tcp syn-flood alarm-threshold
1024
set security screen ids-option screen_web tcp syn-flood attack-threshold
200
set security screen ids-option screen_web tcp syn-flood source-threshold
1024
set security screen ids-option screen_web tcp syn-flood destination-
threshold 2048
set security screen ids-option screen_web tcp syn-flood timeout 20

set security screen ids-option screen_web tcp land
```

4.2.2 screen をゾーンに適用（untrust 推奨）

コマンド（書式）

```
set security zones security-zone <ゾーン名> screen <スクリーン名>
```

何をする？

- 指定ゾーンに対して screen を有効化します（この指定がないと ids-option を作っても効きません）。

例題

```
set security zones security-zone untrust screen screen_web
```

4.2.3 screen の効き具合を確認

コマンド

```
show security screen statistics zone <ゾーン名>
```

何をする？

- 検知/遮断の統計カウンタを表示し、「効いてるか？」を確認します。

例題

```
show security screen statistics zone untrust
```

5. 仕上げ：commit と動作確認（演習の“提出物”の感覚）

5.1 commit

```
commit check
commit
```

- `commit check`：文法・整合性チェック（事故防止）
- `commit`：反映

5.2 動作確認（最低限のチェックリスト）

外部からHTTPが通る（SRX-1 / SRX-2 共通）

- ブラウザで `http://<外部IP>/`
- もしくは `curl http://<外部IP>/`

HTTP以外が拒否される（SRX-2）

- `curl https://<外部IP>/`（443が通らない想定なら失敗する）
- `ssh <外部IP>`（22が通らない想定なら失敗する）

SRX側で「当たったルール」を見る

```
show security policies hit-count
```

DNAT が意図通りか見る

```
show security nat destination rule all
show security nat destination pool
```

パケットが来てるか（最終手段）

```
monitor traffic interface ge-0/0/0.0 no-resolve
```

6. “最小セット”まとめ（SRX-2 を満たす例：コピペ骨格）

> これをあなたの環境の値に置換すると、演習の最短ルートになります。

```
# ゾーン
set security zones security-zone untrust interfaces <UNTRUST_IF>
set security zones security-zone trust   interfaces <TRUST_IF>

# DNAT（外部IP → Web内部IP）
set security nat destination pool pool_web address <WEB_PRIVATE_IP>/32
set security nat destination rule-set 1 from interface <UNTRUST_IF>
set security nat destination rule-set 1 rule web match destination-address
<PUBLIC_OR_OUTSIDE_IP>/32
set security nat destination rule-set 1 rule web then destination-nat pool
pool_web

#  （必要なら）proxy ARP
set security nat proxy-arp interface <UNTRUST_IF> address
<PUBLIC_OR_OUTSIDE_IP>

# ポリシー：HTTPのみ許可
set security policies from-zone untrust to-zone trust policy allow-http
match source-address any
set security policies from-zone untrust to-zone trust policy allow-http
match destination-address any
set security policies from-zone untrust to-zone trust policy allow-http
match application junos-http
set security policies from-zone untrust to-zone trust policy allow-http
then permit

# 明示deny（任意：演習で分かりやすくする）
set security policies from-zone untrust to-zone trust policy deny-rest
match source-address any
set security policies from-zone untrust to-zone trust policy deny-rest
match destination-address any
set security policies from-zone untrust to-zone trust policy deny-rest
match application any
set security policies from-zone untrust to-zone trust policy deny-rest
then deny

# screen（不正検知・抑止）
set security screen ids-option screen_web tcp syn-flood attack-threshold
200
set security zones security-zone untrust screen screen_web
```

7. 便利な確認コマンド（演習で強い）

```
show configuration | display set
show security policies hit-count
show security nat destination rule all
show security screen statistics zone untrust
show security flow session summary
```