Juniper SRXにおける Firewall機能の紹介（概要資料）

目　的

Juniper社製のFirewall（SRX）におけるFirewall機能の概要について紹介します

内　容

１　ファイアウォール(Firewall)の概要
　（1） Firewallとは？
　（2） Firewallによるフィルタリング方式
　（3） ステートフルインスペクション形FWの動作
　（4） コネクションテーブルを確認してみよう！
２　Firewallルールの概要と設定
　（1） ルールの概要
　（2） Objectの作成
　（3） ルールの作成、適用
　（4） ルール適用例
　（5） 通信中のセッションに対する新規ルールの適用
３　参考資料
プリデファインアプリケーションの設定と確認
参考資料
　Junosハンズオントレーニング　SRXシリーズサービスゲートウェイ　　【Juniper　Networks社資料】
　JunosにおけるFirewall機能について
　　https://www.juniper.net/assets/jp/jp/local/pdf/others/firewall_configuration.pdf

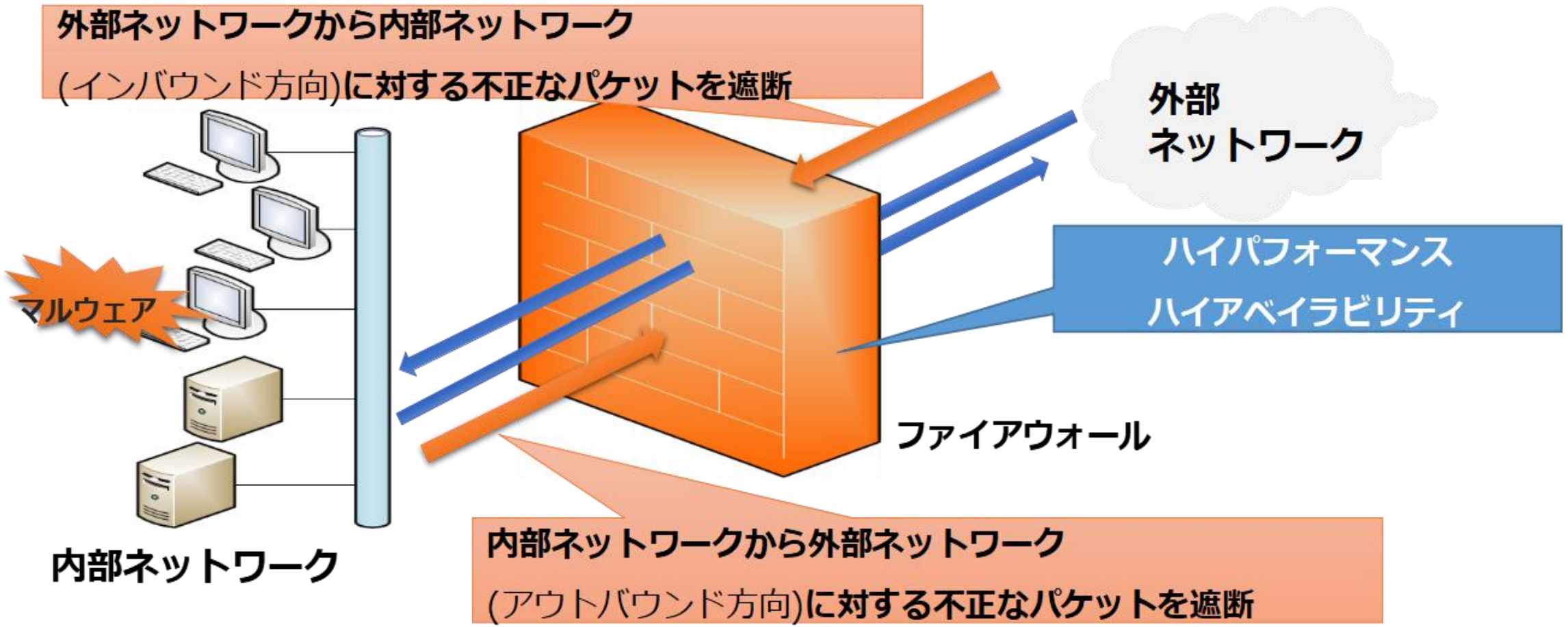
１　ファイアウォール(FW)の概要

（1）　ファイアウォールとは？？

1　ファイアウォール(FW)の概要

(1) ファイアウォールとは？？

内部のコンピュータネットワークと外部との通信を制御し、内部のコンピュータネットワークの安全を維持することを目的としたソフトウェアの技術概念



1　ファイアウォール(FW)の概要

（2） ファイアウォールによるフィルタリング方式

1　ファイアウォール(FW)の概要

(2) ファイアウォールによるフィルタリング方式

フィルタリングの方式は以下の3つになります。

① パケットフィルタリング型（ステートレス）

パケットのヘッダ情報に含まれるIPアドレス、ポート番号に基づいてフィルタリングを行う

② アプリケーションレベルゲートウェイ型

プロトコルごとにプロキシ(中継専用プログラム)をもち、パケットのアプリケーション層も含めた情報に基づいてフィルタリングを行う

現在、主流

③ ステートフルインスペクション型（ステートフル）

セッション(通信の開始から終了まで管理する単位)の状態を管理して、常にその情報に基づいてフィルタリングを行う。

1　ファイアウォール(FW)の概要

（3） ステートフルインスペクション方式FWの動作

1　ファイアウォール(FW)の概要

(3)　ステートフルインスペクション型FWの動作

フィルタリングルールとコネクションテーブル双方が連携して動作

フィルタリングルールとコネクションテーブルとは？？

【フィルタリングルール】:管理者が設定
　どんな通信を許可し、どんな通信を拒否するかを定義している設定です。
　　設定項目:送信元IPアドレス、宛先IPアドレス、プロトコル、送信元ポート番号、
　　通信制御　などがあります。

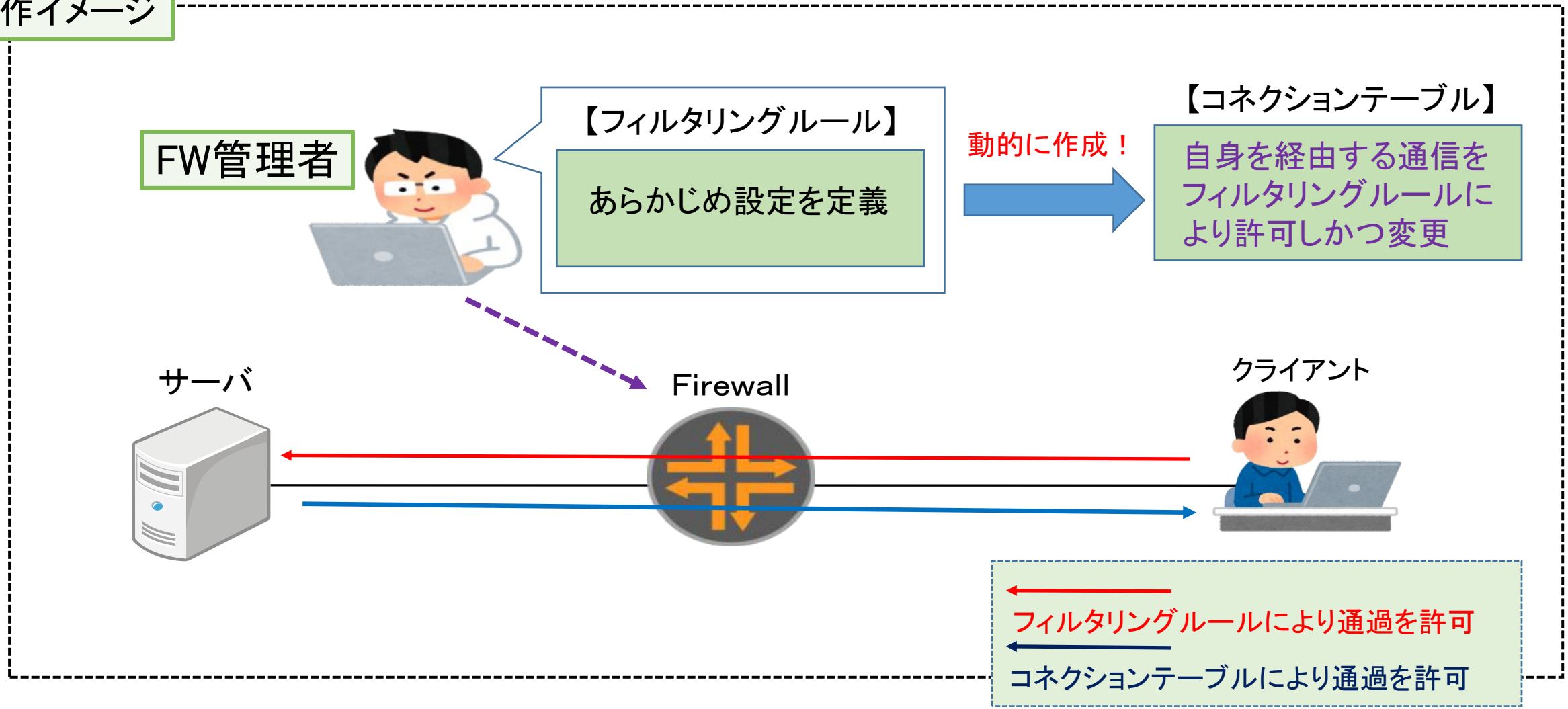
【コネクションテーブル】:通過する通信により動的に作成
　自身を経由するコネクションの情報を管理しているテーブル
　　管理項目:
　　送信元IPアドレス、宛先IPアドレス、プロトコル、送信元IPポート番号、
　　宛先ポート番号、コネクションの状態、アイドルタイムアウト　などがあります。

1　ファイアウォール(FW)の概要

(3)　ステートフルインスペクション型FWの動作

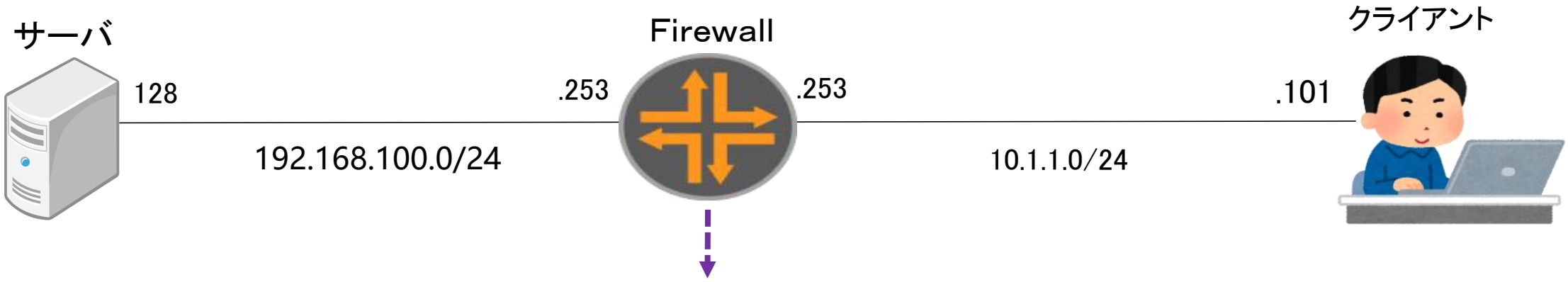
フィルタリングルールとコネクションテーブル双方が連携して動作

動作イメージ



1　ファイアウォール(FW)の概要

(3)　ステートフルインスペクション型FWの動作



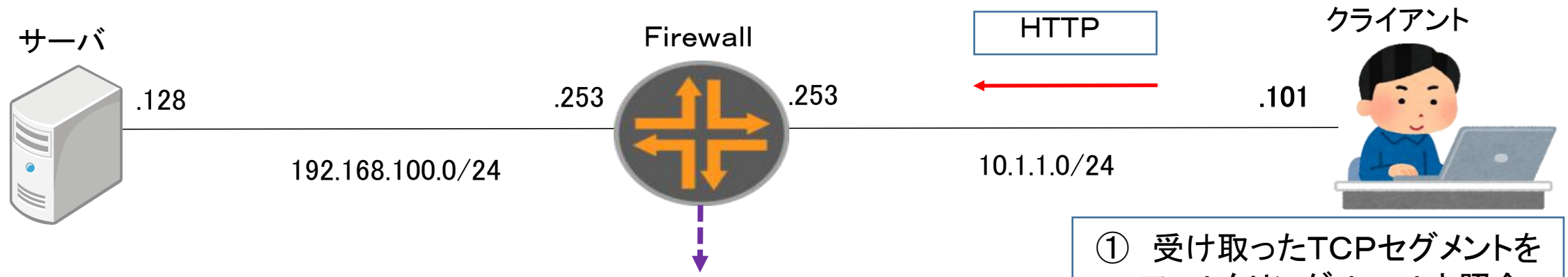
フィルタリングルール					
送信元IP	宛先IP	プロトコル	送信元ポート	宛先ポート	アクション
10.1.1.0/24	192.168.100.128	TCP	Any	80	許可
10.1.1.0/24	192.168.100.128	TCP	Any	22	拒否
10.1.1.0/24	192.168.100.128	TCP	Any	23	ドロップ

管理者

フィルタリングルール
はあらかじめ管理者が作成し、
設定する

1　ファイアウォール(FW)の概要

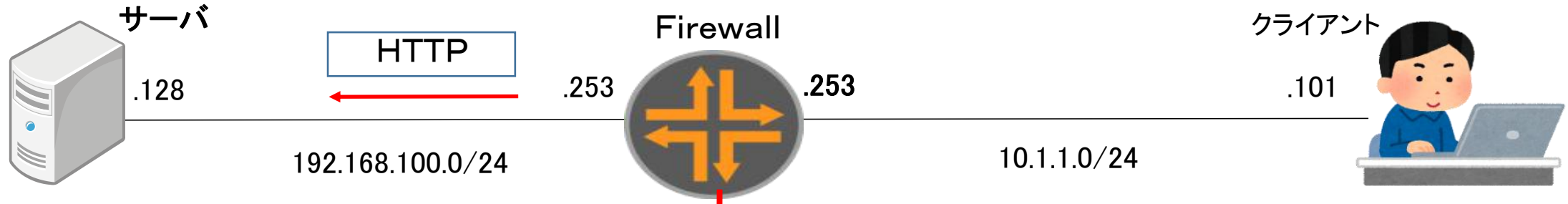
(3)　ステートフルインスペクション型FWの動作



フィルタリングルール					
送信元IP	宛先IP	プロトコル	送信元ポート	宛先ポート	アクション
10.1.1.0/24	192.168.100.128	TCP	Any	80	許可
10.1.1.0/24	192.168.100.128	TCP	Any	22	拒否
10.1.1.0/24	192.168.100.128	TCP	Any	23	ドロップ

1　ファイアウォール(FW)の概要

(3)　ステートフルインスペクション型FWの動作



② 許可エントリに合致したら
コネクションエントリを作る

フィルタリングルール

送信元IP	宛先IP	プロトコル	送信元ポート	宛先ポート	アクション
10.1.1.0/24	192.168.100.128	TCP	Any	80	許可
10.1.1.0/24	192.168.100.128	TCP	Any	22	
10.1.1.0/24	192.168.100.128	TCP	Any	23	
192.168.100.128	10.1.1.101	TCP	80	59226	許可

③ コネクションエントリから
戻り通信用の許可エントリ

コネクションテーブル

送信元IP	宛先IP	プロトコル	送信元ポート	宛先ポート	状態	アイドルタイム
10.1.1.0/24	192.168.100.128	TCP	Any	80	SYN	0秒

1　ファイアウォール(FW)の概要

（4） コネクションテーブルを確認してみよう！

1　ファイアウォール(FW)の概要

(4) コネクションテーブルを確認してみよう！

イメージ

【コネクションテーブル確認コマンド】
show security flow session

サーバ

Juniper　SRX

クライアント

コネクションテーブル
（通信状態を管理）

ターミナルソフト
（Putty）

管理端末

1　ファイアウォール(FW)の概要

(4) コネクションテーブルを確認してみよう！

>show security flow session

```
lab> show security flow session Session

ID: 6352, Policy name: self-traffic-policy/1, Timeout: 1800, Valid
 In: 172.27.3.149/2462 --> 172.27.67.101/23;tcp, If: fe-0/0/0.0, Pkts: 2156, Bytes: 87809
Out: 172.27.67.101/23 --> 172.27.3.149/2462;tcp, If: .local..0, Pkts: 1478, Bytes: 140513
Session ID: 6565, Policy name: trust-to-untrust/4, Timeout: 1788, Valid
In: 192.168.1.254/2330 --> 207.46.73.60/80;tcp, If: vlan.0, Pkts: 17, Bytes: 2216
Out: 207.46.73.60/80 --> 172.27.67.101/18983;tcp, If: fe-0/0/0.0, Pkts: 19, Bytes: 23973
Session ID: 6991, Policy name: trust-to-untrust/4, Timeout: 1774, Valid
In: 192.168.1.254/2298 --> 64.233.183.138/80;tcp, If: vlan.0, Pkts: 10, Bytes: 3102
Out: 64.233.183.138/80 --> 172.27.67.101/6412;tcp, If: fe-0/0/0.0, Pkts: 7, Bytes: 121241
Session ID: 6998, Policy name: self-traffic-policy/1, Timeout: 4, Valid
In: 172.27.67.101/123 --> 210.173.160.87/123;udp, If: .local..0, Pkts: 1, Bytes: 76
Out: 210.173.160.87/123 --> 172.27.67.101/123;udp, If: fe-0/0/0.0, Pkts: 1, Bytes: 76
Session ID: 6999, Policy name: self-traffic-policy/1, Timeout: 10, Valid
In: 172.27.67.101/123 --> 210.173.160.57/123;udp, If: .local..0, Pkts: 1, Bytes: 76
Out: 210.173.160.57/123 --> 172.27.67.101/123;udp, If: fe-0/0/0.0, Pkts: 1, Bytes: 76
```

２　Firewallルールの概要と設定

（1）　ルールの概要

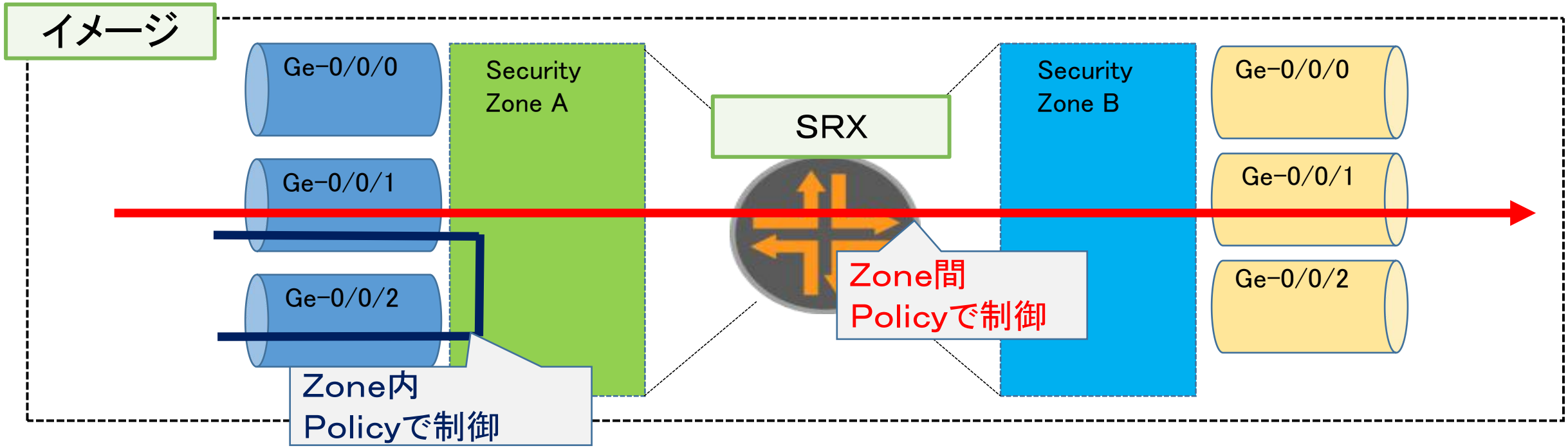
1　ファイアウォールルールの概要と設定

(1) ルールの概要

Security ZoneとSecurity Policyにより通過するトラヒックを制御します！！

Security Zone
インタフェースに割り当てる仮想的なグループ（SRXではZoneを使用してトラヒックを制御）

Security Policy
SRXを通過するトラヒックを制御するためのルール
（Zone間トラヒックと、Zone内トラヒックにそれぞれ適用されます）



1　ファイアウォールルールの概要と設定

(1) ルールの概要

【ルール反映までの流れ】

Objectの作成

対象となるアドレス、アプリケーション情報を作成します。
（address-book/address-set）　→　any設定する場合は必要ありません。。

↓

Ruleの作成、適用

通信許可/遮断に関するRuleの追加等を実施します。
（firewall-policy）

↓

設定の反映

作成したRuleを保存します。
（commit）

注意！：commitしないと
設定が反映されません

ZoneとPolicy設定については設定済みとします。。

２　Firewallルールの概要と設定

（2）　Objectの作成

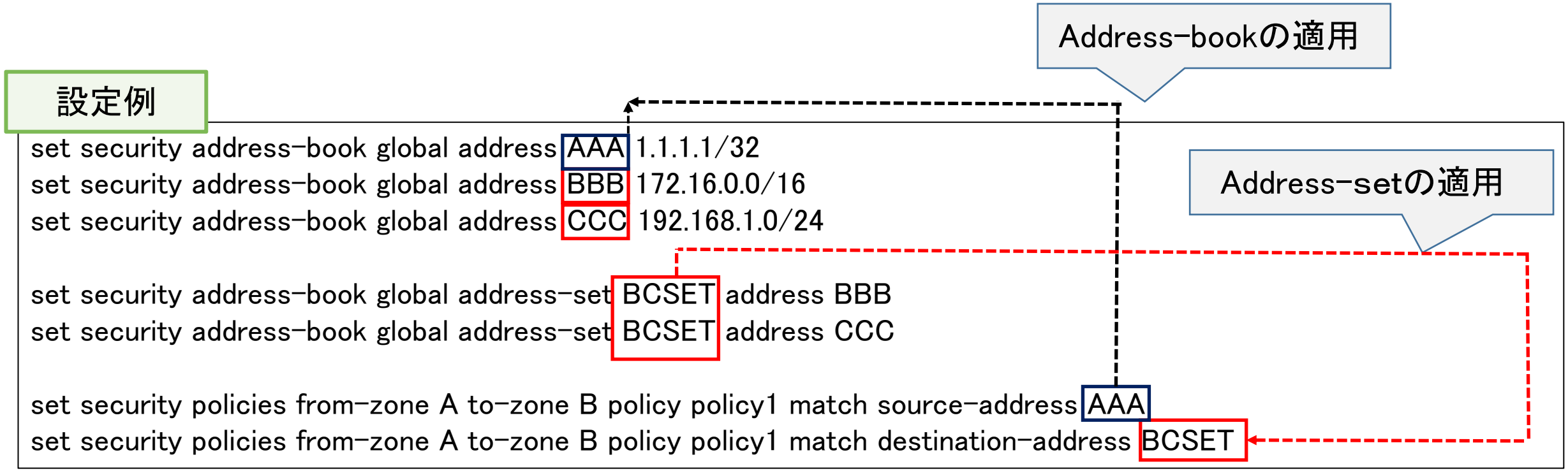
2　Firewallルールの概要と設定

(2)　Objectの作成

ルール適用時の対象サブネット(ホスト)等を指定する際に使用
(address-bookを複数組み合わせて使用する場合にはaddress-setを使用)

《設定例》
異なるIPアドレス(サブネット)を複数指定してFirewallポリシーに適用したい。

ADDRESS BOOKの定義　1.1.1.1/32　⇒　AAA　172.16.0.0/16　⇒　BBB　192.168.1.0/24　⇒　CCC
ADDRESS SETの定義　BCSET　⇒　BBB、CCCを組み合わせ



2　Firewallルールの概要と設定

(2)　Objectの作成

ルール適用時の対象サブネット(ホスト)等を指定する際に使用
（address-bookを複数組み合わせて使用する場合にはaddress-setを使用）

address-book、address-setの適用については以下の2つの方法があります。

1　装置全体で適用する場合

```
set security address-book global address AAA 1.1.1.1/32
set security address-book global address BBB 172.16.0.0/16
set security address-book global address CCC 192.168.1.0/24
```

2　個別のSecurity zoneで適用する場合

```
set security  zones security-zones untrust  address-book address AAA 1.1.1.1/32
set security  zones security-zones untrust  address-book address BBB  172.16.0.0/16
set security  zones security-zones untrust  address-book address CCC 192.168.1.0/24
```

２　Firewallルールの概要と設定

（3）　ルールの作成/適用

2　Firewallルールの概要と設定

(3)　ルールの作成/適用

MatchとThenでトラヒックを評価してアクションを決定

Match　ー　Policyに合致させる条件を設定する

Then　ー　条件に合致した通信に対するアクション(動作)を設定する

《設定条件》

Trust Zone ⇒ UnTrust Zone Policy				
Policy名	送信元	宛先	アプリケーション	判定
Policy1	any	Any	Any	許可(Permit)

match

then

source-address　：送信元
destination-address　：宛先
apptication　：アプリケーション

permit　：許可
deny　：廃棄(エラーを返さない)
reject　：拒否(エラーコードを返す)
log　：ログを残す
Count　：該当ポリシーのパケット数、バイト数を取得

```
set security policies from-zone trust to-zone untrust policy policy1 match source-address any
set security policies from-zone trust to-zone untrust policy policy1 match destination-address any
set security policies from-zone trust to-zone untrust policy policy1 match application any
set security policies from-zone trust to-zone untrust policy policy1 then permit
```

設定例

2　Firewallルールの概要と設定

(3)　ルールの作成/適用

Firewallルールの修正について

Firewallルールは最初の行から処理されますので。。

新規にルールを追加する場合は　“insert”コマンドを使用します。

（設定例）

【新規に適用したいルールを設定します】

```
set security address-book global address UNTRUST_ADD-0001 XXX.XXX.XXX.XXX/XX
set security address-book global address-set UNTRUST_ADD_SET-0001 address UNTRUST_ADD-0001
set security policies from-zone TRUST to-zone UNTRUST policy TRUST_UNTRUST_POLICY-0001 match
source-address any
set security policies from-zone TRUST to-zone UNTRUST policy TRUST_UNTRUST_POLICY-0001
match destination-address UNTRUST_ADD_SET-0001
set security policies from-zone TRUST to-zone UNTRUST policy TRUST_UNTRUST_POLICY-0001
match application any
set security policies from-zone TRUST to-zone UNTRUST policy TRUST_UNTRUST_POLICY-0001 then permit
```

【現行のルールの前に設定したルールを追加します】

```
insert security policies from-zone TRUST to-zone UNTRUST policy TRUST_UNTRUST_POLICY-0001 before
policy ALL_DENY
```

２　Firewallルールの概要と設定

（4）　ルール適用例

2　Firewallルールの概要と設定

(4)　ルール適用例

以下の状況におけるFirewallルール適用例について紹介します。

○　定義済みアプリケーションを指定してPOLICYに適用する場合（その1、その2）
　Well-knowポートを使用するアプリケーション（HTTPなど）に適用するとき

○　新たなアプリケーションポート番号を指定し、POLICYに適用する場合
　独自アプリケーションなど普段使用しないポート番号に適用するとき

○　指定すべきアドレス範囲が点在している場合
　アドレスブックで指定したいアドレス範囲が点在している。。際に適用する場合

２　Firewallルールの概要と設定

（4）　ルール適用例

○　定義済みアプリケーションを指定してPOLICYに適用する場合（その１、その２）

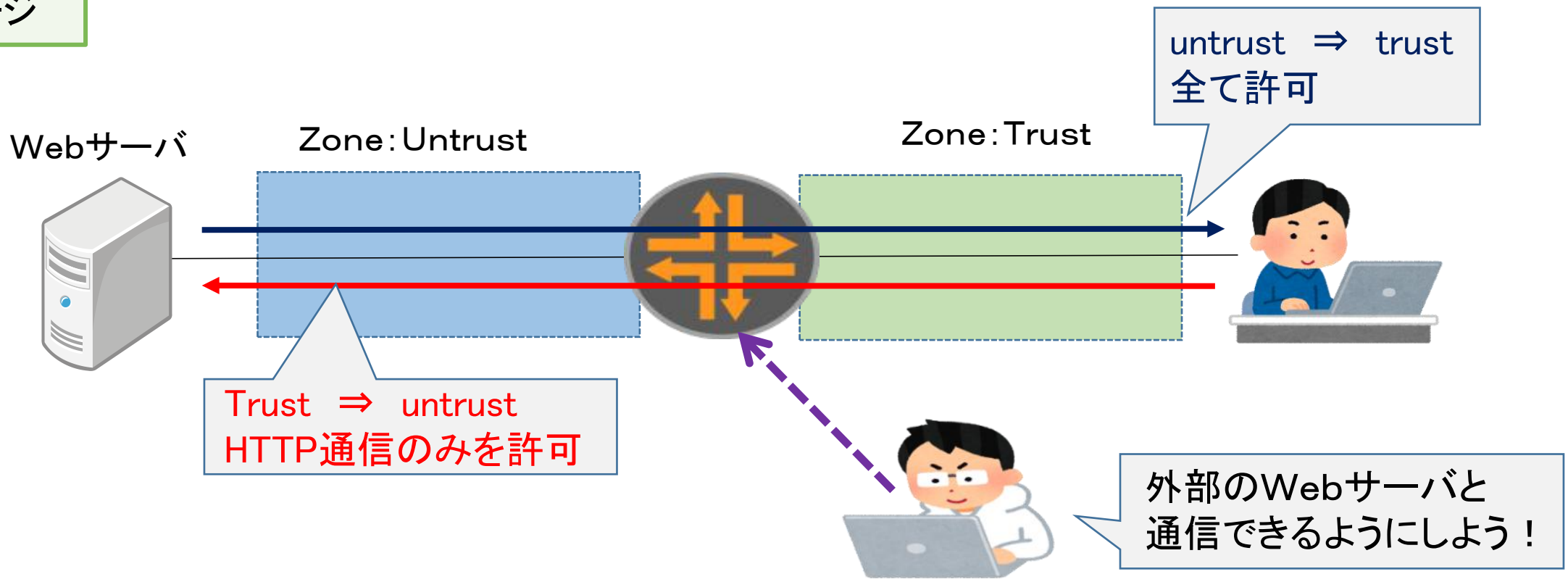
2　Firewallルールの概要と設定

(4)　ルール適用例

○　定義済みアプリケーションを指定してPOLICYに適用する場合（その1）

【例：HTTP通信のみを許可するルールを作成したい！】

イメージ



2　Firewallルールの概要と設定

(4)　ルール適用例

○　定義済みアプリケーションを指定してPOLICYに適用する場合（その1）

【例：HTTP通信のみを許可するルールを作成したい！】

定義済みアプリケーション
（HTTPのTCP/80を指定）

設定例

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

Policy名：HTTP-Permit

```
set security policies from-zone untrust to-zone trust policy all-permit match source-address any
Set security policies from-zone untrust to-zone trust policy all-permit match destination-address any
set security policies from-zone untrust to-zone trust policy all-permit match application any
Set security policies from-zone trust to-zone untrust policy all-permit then permit
Set security policies default-polices deny-all
```

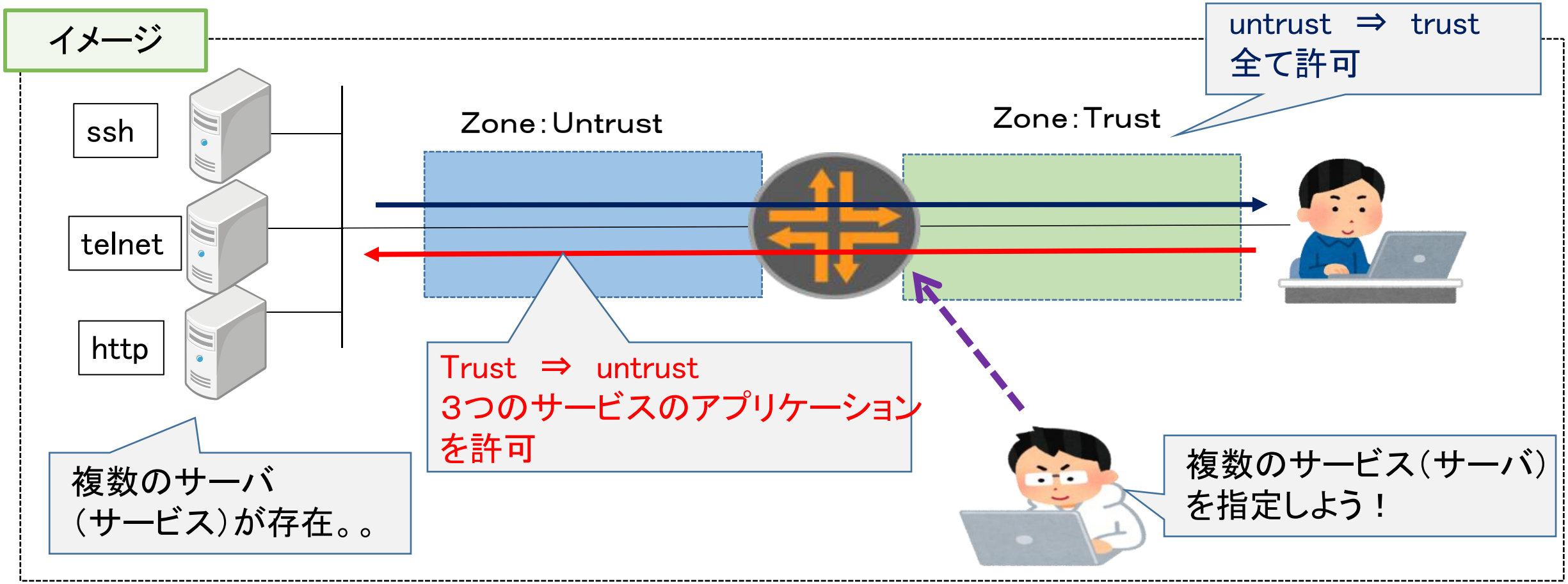
／　デフォルトポリシー（廃棄）

2　Firewallルールの概要と設定

(4)　ルールの適用例

○　定義済みアプリケーションを指定してPOLICYに適用する場合（その1）

【例：複数のアプリケーションをまとめて指定したい！】



2　Firewallルールの概要と設定

(4)　ルール適用例

○　定義済みアプリケーションを指定してPOLICYに適用する場合（その1）

【例：複数のアプリケーションをまとめて指定したい！】

複数のサーバ
（サービス）を選択

設定例

```
set applications application-set MANAGE-SET application junos-ssh
set applications application-set MANAGE-SET application junos-telnet
set applications application-set MANAGE-SET application junos-telnet

set security policies from-zone trust to-zone untrust policy Server-permit match source-address any
set security policies from-zone trust to-zone untrust policy Server-permit match destination-address any
set security policies from-zone trust to-zone untrust policy Server-permit match application MANAGE-SET
Set security policies from-zone trust to-zone untrust policy Server-permit then permit
set security policies from-zone untrust to-zone trust policy all-permit match source-address any
set security policies from-zone untrust to-zone trust policy all-permit match destination-address any
set security policies from-zone untrust to-zone trust policy all-permit match application any
Set security policies from-zone trust to-zone untrust policy all-reject then permit
Set security policies default-polices deny-all                    /　デフォルトポリシー（廃棄）
```

２　Firewallルールの概要と設定

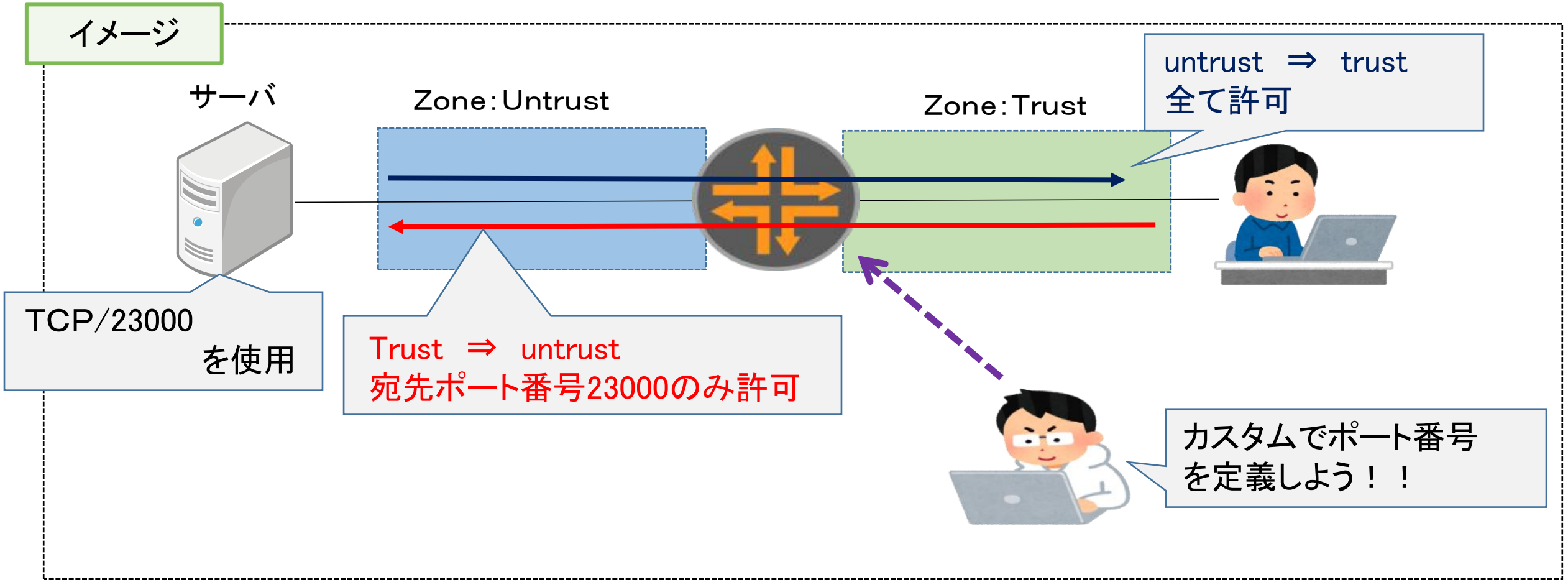
（4）　ルール適用例

〇新たなアプリケーションポート番号を指定し、POLICYに適用する場合

2　Firewallルールの概要と設定

(4)　ルール適用例

〇新たなアプリケーションポート番号を指定し、POLICYに適用する場合
【例：通常使用しないポート番号（Well-Knownポート以外）を指定したい！】



2　Firewallルールの概要と設定

(4)　ルール適用例

○新たなアプリケーションポート番号を指定し、POLICYに適用する場合
【例：通常使用しないポート番号（Well-Knownポート以外）を指定したい！】

TCP/23000を定義

設定例

```
set applications application untrust-Server protocols tcp source-port 1-65535 destination-port 23000

set security policies from-zone trust to-zone untrust policy Server-permit match source-address any
set security policies from-zone trust to-zone untrust policy Server-permit match destination-address any
set security policies from-zone trust to-zone untrust policy Server-permit match application untrust-Server
Set security policies from-zone trust to-zone untrust policy Server-permit then permit
set security policies from-zone untrust to-zone trust policy all-permit match source-address any
set security policies from-zone untrust to-zone trust policy all-permit match destination-address any
set security policies from-zone untrust to-zone trust policy all-permit match application any
Set security policies from-zone trust to-zone untrust policy all-permit then permit
Set security policies default-polices deny-all                    /　デフォルトポリシー（廃棄）
```

２　Firewallルールの概要と設定

（4）　ルール適用例

○　　指定すべきアドレス範囲が点在している場合。。

2　Firewallルールの概要と設定

(4)　ルール適用例

○　指定すべきアドレス範囲が点在している場合。。

イメージ

攻撃者（その1）

IP：130.135.110.101
～110

攻撃者（その2）

IP：130.190.100.100
～103

WAN

Zone：Untrust

Zone：Trust

utrust側で遮断

複数拠点から攻撃。。
効率的に設定追加したい。。

攻撃者が複数ホストかつ
複数拠点から攻撃！

2　Firewallルールの概要と設定

(4)　ルール適用例

○　指定すべきアドレス範囲が点在している場合。。

Address-book/Address-set
を追加するだけで良い！！

設定例

アドレスレンジを指定

```
set security address-book global address Attaker-1 range-address 130.135.110.101 to 130.135.110.110
set security address-book global address Attaker-2 range-address 130.190.100.100 to 130.190.100.103

set security address-book global address-set Attaker-segment address Attaker-1
set security address-book global address-set Attaker-segment address Attaker-2

set security policies from-zone untrust to-zone trust policy Attaker-deny match source-address Attaker-segment
set security policies from-zone untrust to-zone trust policy Attaker-deny match destination-address any
set security policies from-zone untrust to-zone trust policy Attaker-deny match application any
Set security policies from-zone untrust to-zone trust policy Attaker-deny then deny

Set security policies default-polices permit-all
```

address-setに割り当て

／　デフォルトポリシー（通過）

4　Firewallルールの概要と設定

（5） 通信中のセッションに対する新規ルールの適用

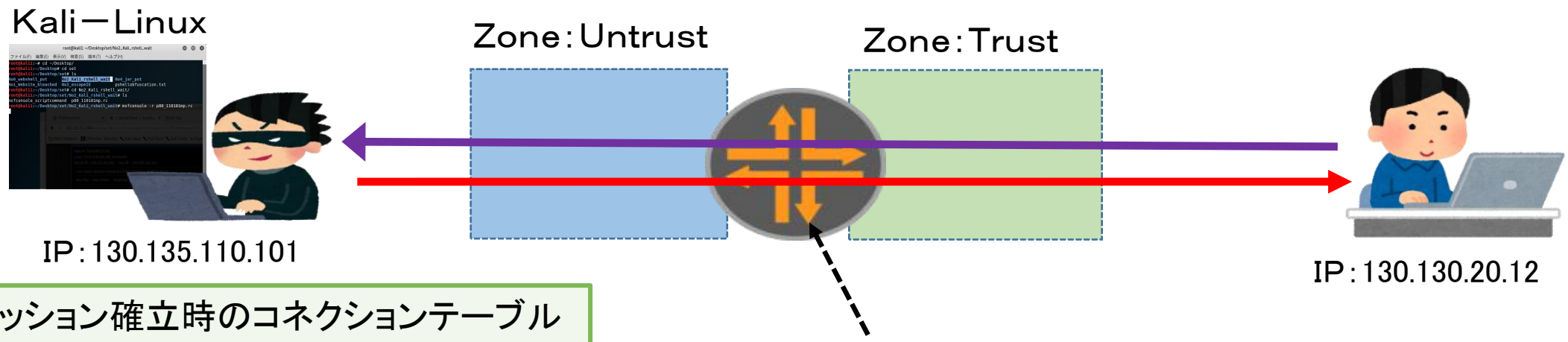
2　Firewallルールの概要と設定

(5) 通信中のセッションに対する新規ルールの適用

通信中のセッションに対して新たにルールを適用して切断したい！

→　　ルールを適用するには各種条件があります。。。

例） アタッカーとセッションを確立



セッション確立時のコネクションテーブル

```
show security flow session | match 130.135.110.101 | match 443
  In: 130.130.20.12/59803 --> 130.135.110.101/443;tcp, Conn Tag: 0x0, If: ge-0/0/5.0,Pkts: 52, Bytes: 32555,
  Out: 130.135.110.101/443 --> 130.130.20.12/59803;tcp, Conn Tag: 0x0, If: ge-0/0/4.0, Pkts: 57, Bytes: 3289,
```

新規のルールを設定

既存のセッションを切断！

2　Firewallルールの概要と設定

(5) 通信中のセッションに対する新規ルールの適用

〇Policy-rematchフラグの追加について

通信中のセッションに対して新規ルールを適用する場合は
事前にPolicy-rematchフラグの設定が必要です！

Policy-rematchフラグが有効の場合におけるJunos OSが実行するアクションは以下の通りです。

- ポリシーを挿入する：　影響なし
- ポリシーのActionフィールドをpermitからdenyまたはrejectのいずれかに変更する
  ：既存のセッションがすべてドロップする
  （すべてのセッションがドロップするためお勧めしません）
- 送信元アドレス、宛先アドレス、アプリケーションフィールドの一部の組み合わせを変更する
  ：Junos OSがポリシールックアップを再評価する・・
  （こちらの方がベスト！）

2　Firewallルールの概要と設定

(5)　通信中のセッションに対する新規ルールの適用

○Policy-rematchフラグの追加について

Policy-Rematchフラグにおける Junos OSが実行するアクション一覧

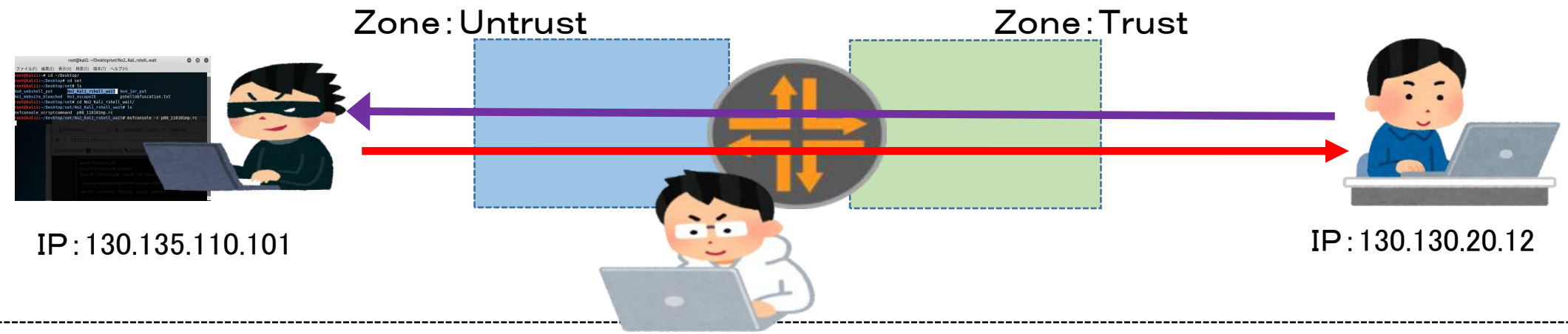
ポリシーのアクション	説明	Rematchフラグ	
		有効	無効（デフォルト）
削除	ポリシーを削除する	既存のセッションを すべてドロップする	既存のセッションを すべてドロップする
アクションの変更	ポリシーのActionフィールドを PermitからDenyまたは Rejectに変更する	既存のセッションを すべてドロップする	既存のセッションを すべて削除する
アドレスの変更	送信元または宛先アドレスを 変更する	ポリシールックアップを 再評価する	既存のセッションを すべて継続する
アプリケーションの変更	アプリケーションを変更する	ポリシールックアップを 再評価する	既存のセッションをすべて 継続する

細部は　SRXシリーズ　スタディガイド　ーパート1　第3章　セキュリティポリシー　を参照

2　Firewallルールの概要と設定

(5) 通信中のセッションに対する新規ルールの適用

ー　ポリシーのActionフィールドをpermitからdenyまたはrejectのいずれかに変更する



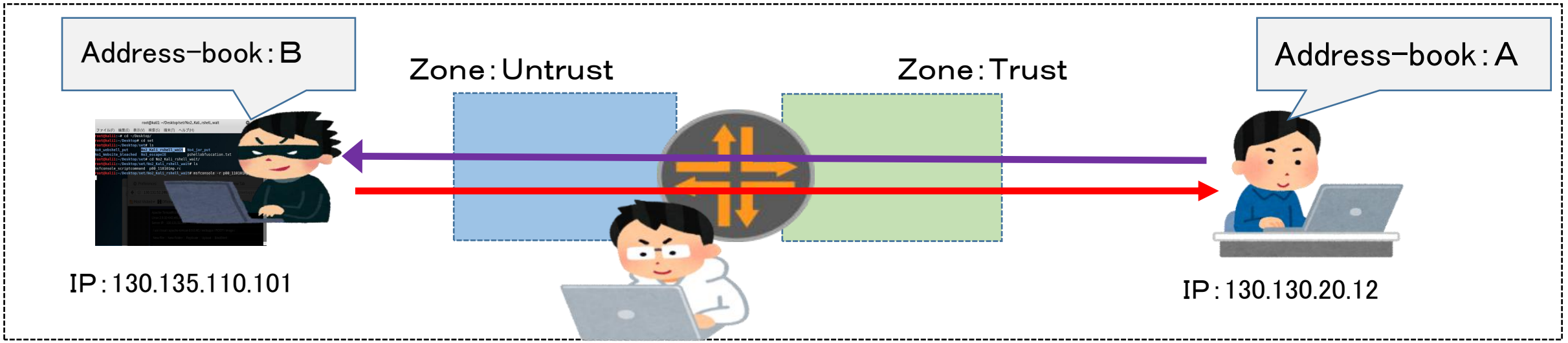

```
set security policies from-zone trust to-zone untrust policy trust_to_untrust  source-address any
set security policies from-zone trust to-zone untrust policy trust_to_untrust  match destination-address any
set security policies from-zone trust to-zone untrust policy trust_to_untrust  match application any
Set security policies from-zone trust to-zone untrust policy trust_to_untrust  then deny
set security policies from-zone untrust to-zone trust policy untrust_to_trust source-address any
set security policies from-zone untrust to-zone trust policy untrust_to_trust match destination-address any
set security policies from-zone untrust to-zone trust policy untrust_to_trust match application any
Set security policies from-zone trust to-zone untrust policy untrust_to_then deny
Set security policies polices-rematch                                    /ポリシーリマッチを適用
```

2　Firewallルールの概要と設定

(5)　通信中のセッションに対する新規ルールの適用

お勧めのやり方！

一送信元アドレス、宛先アドレス、アプリケーションフィールドの一部の組み合わせを変更する

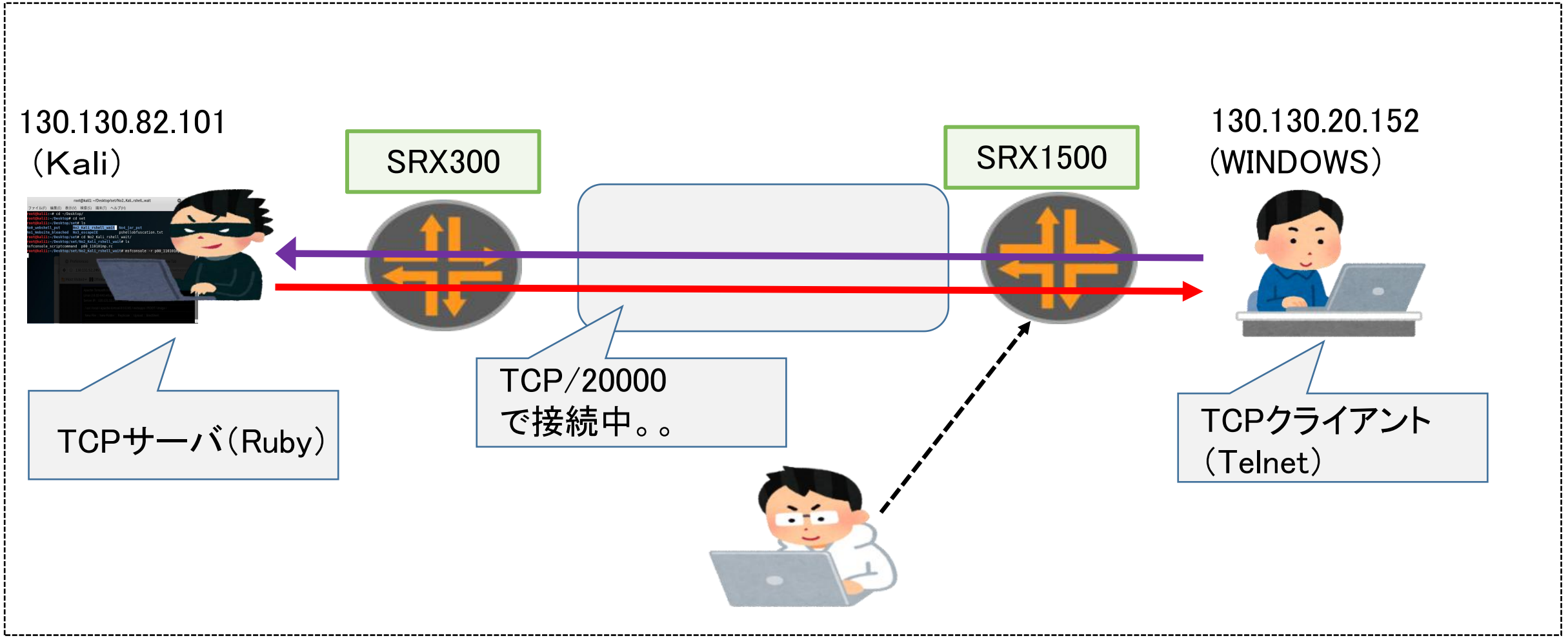



```
set security policies from-zone trust to-zone untrust policy trust_to_untrust source-address A
set security policies from-zone trust to-zone untrust policy trust_to_untrust match destination-address B
set security policies from-zone trust to-zone untrust policy trust_to_untrust match application any
Set security policies from-zone trust to-zone untrust policy trust_to_untrust then deny
set security policies from-zone untrust to-zone trust policy untrust_to_trust source-address B
set security policies from-zone untrust to-zone trust policy untrust_to_trust match destination-address any
set security policies from-zone untrust to-zone trust policy untrust_to_trust match application any
Set security policies from-zone trust to-zone untrust policy untrust_to_trust then deny
Set security policies polices-rematch                                    /ポリシーリマッチを適用
```

2　Firewallルールの概要と設定

(5) 通信中のセッションに対する新規ルールの適用

○展示



SRX1500でルールを変更し通信中のセッションを切断します。。

2　Firewallルールの概要と設定

(5) 通信中のセッションに対する新規ルールの適用

○展示

SRX1500の設定

```
/アドレスブックの設定
set security address-book YONA 130.132.82.101/32
set security address-book ICHI  130.130.20.153/24



set security policies from-zone trust to-zone untrust policy trust_to_untrust source-address ICHI
set security policies from-zone trust to-zone untrust policy trust_to_untrust match destination-address YONA
set security policies from-zone trust to-zone untrust policy trust_to_untrust match application any
Set security policies from-zone trust to-zone untrust policy trust_to_untrust then reject
set security policies from-zone untrust to-zone trust policy untrust_to_trust source-address YONA
set security policies from-zone untrust to-zone trust policy untrust_to_trust match destination-address ICHI
set security policies from-zone untrust to-zone trust policy untrust_to_trust match application any
Set security policies from-zone trust to-zone untrust policy untrust_to_trust then reject
Set security policies polices-rematch                                    /ポリシーリマッチを適用
```

端末側にエラーメッセージを送付するためrejectを設定

→ 仮想端末から通信した場合はReject設定によりSRXからICMPの到達不能メッセージが返信され端末のコネクションが切断されます。。

５　参考資料

プリデファインアプリケーション設定の確認

5　参考資料

プリデファインアプリケーション設定の確認

configモードで以下のコマンドにより確認します
root#show groups junos-defaults applications

”Junos-〇〇”に対する
ポート番号が予め設定されています

```
[edit]
admin# show groups junos-defaults applications
##
## protect: groups junos-defaults
##
#
# File Transfer Protocol
#
application junos-ftp {
    application-protocol ftp;
    protocol tcp;
    destination-port 21;
}
application junos-ftp-data {
    application-protocol ftp-data;
    protocol tcp;
}
#
# Trivial File Transfer Protocol
#
```

5　参考資料

プリデファインアプリケーション設定の確認

各アプリケーションにおける
タイムアウト値が設定されている

TCPアプリケーションのタイムアウト値を確認する場合

root > request pfe execute target fpc0 command "show usp app-def tcp"

```
root> request pfe execute target fpc0 command "show usp app-def tcp"
================== tnp_0x10000080 ==================
SENT: Ukern command: show usp app-def tcp

tcp port=0, appl_name=junos-tcp-any, service type=0, alg id=0, timeout=1800
tcp port=21, appl_name=junos-ftp, service type=1, alg id=1, timeout=1800
tcp port=22, appl_name=junos-ssh, service type=22, alg id=0, timeout=1800
tcp port=23, appl_name=junos-telnet, service type=10, alg id=0, timeout=1800
tcp port=25, appl_name=junos-smtp, service type=7, alg id=0, timeout=1800
tcp port=43, appl_name=junos-whois, service type=46, alg id=0, timeout=1800
tcp port=49, appl_name=junos-tacacs, service type=0, alg id=0, timeout=1800
tcp port=53, appl_name=junos-dns-tcp, service type=16, alg id=16, timeout=1800
tcp port=65, appl_name=junos-tacacs-ds, service type=0, alg id=0, timeout=1800
tcp port=70, appl_name=junos-gopher, service type=39, alg id=0, timeout=1800
tcp port=79, appl_name=junos-finger, service type=17, alg id=0, timeout=1800
tcp port=80, appl_name=junos-http, service type=6, alg id=0, timeout=1800
tcp port=110, appl_name=junos-pop3, service type=8, alg id=0, timeout=1800
tcp port=111, appl_name=junos-sun-rpc-tcp, service type=5, alg id=5, timeout=240
0
tcp port=113, appl_name=junos-ident, service type=34, alg id=0, timeout=1800
tcp port=119, appl_name=junos-nntp, service type=35, alg id=0, timeout=1800
tcp port=135, appl_name=junos-ms-rpc-tcp, service type=55, alg id=55, timeout=18
00
tcp port=139, appl_name=junos-smb, service type=21, alg id=0, timeout=1800
---(more)---
```