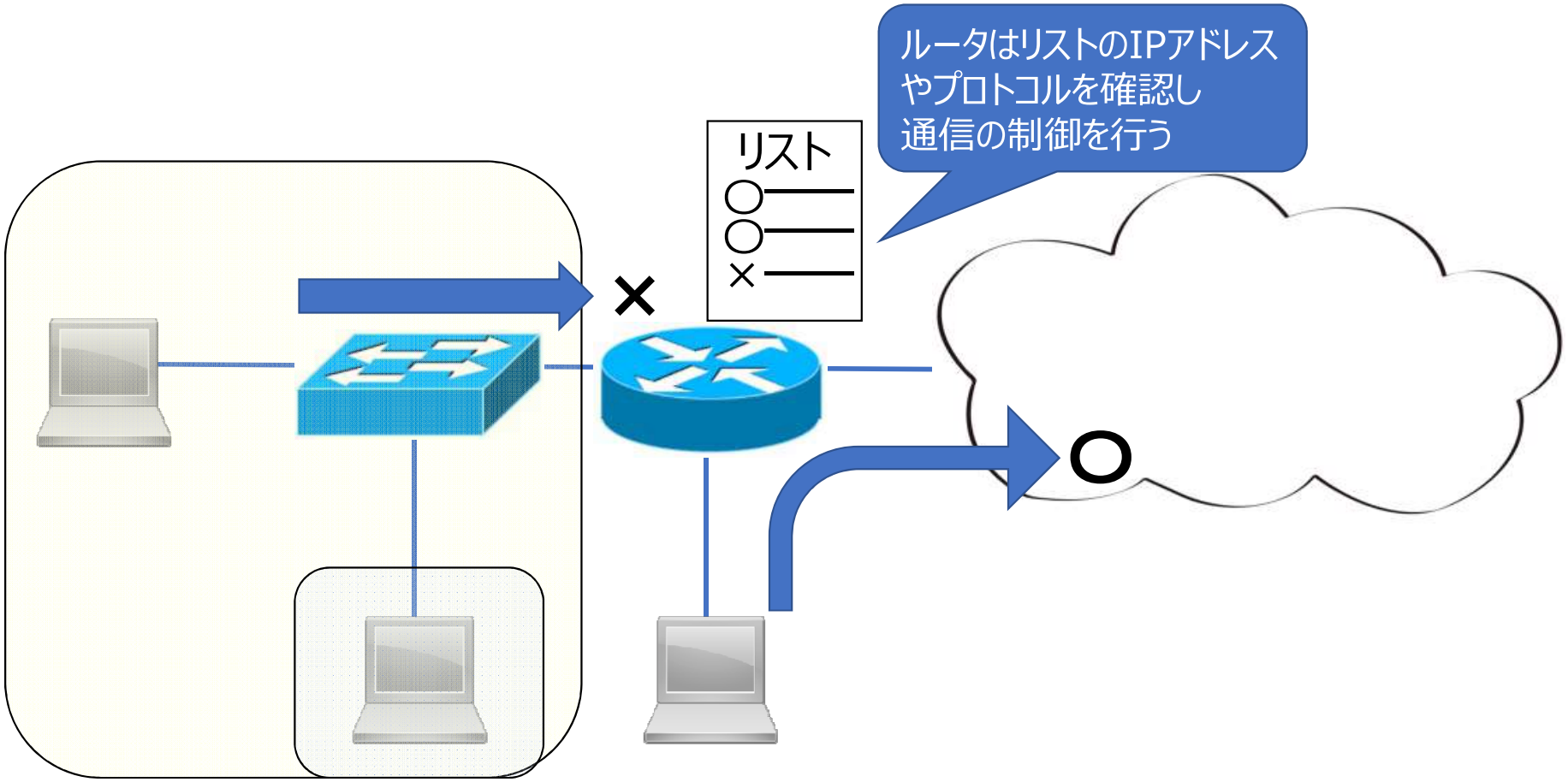
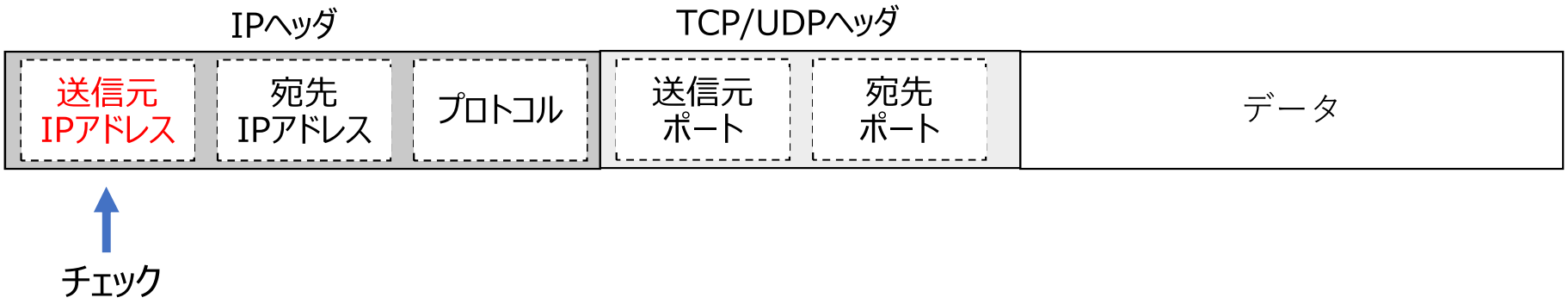
パケットフィルタリング
ACL(Access Control List)

ACL

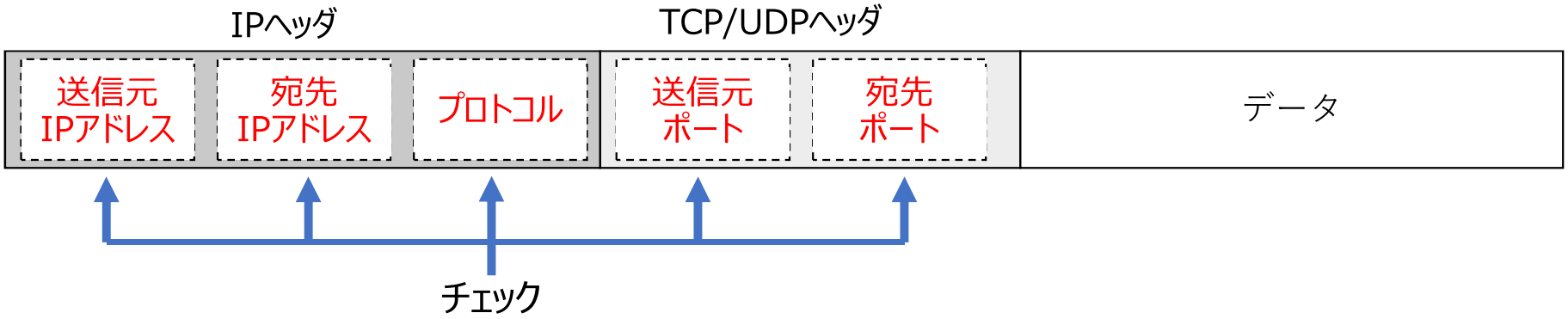
リストにより通信出来るデータと通信させない出来ないデータを登録する



・標準ACL
パケットのIPヘッダ内にある送信元IPアドレスのみチェックされ、パケット転送の許可または拒否が決定



・拡張ACL
条件として送信元IPアドレスと宛先IPアドレス、プロトコル、送信元ポート番号、宛先ポート番号も指定可能
拡張ACLでは、複数の条件に基づいてパケット転送の許可または拒否が決定されるため、より柔軟で複雑な制御が可能

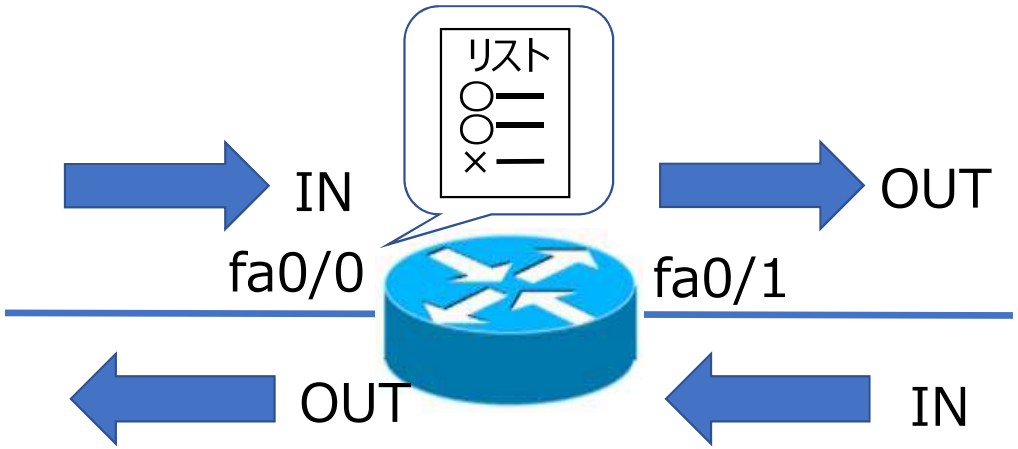


ACLの適用

インバウンド、アウトバウンドにそれぞれ1つIPv4、IPv6のACLを登録可能（この構成図の場合、最大8個設定可能）

インバウンド　　ACL照合　⇒　ルーティングテーブル
アウトバウンド　ルーティングテーブル　⇒　ACL照合

※インバウンドACLは拒否されたパケットのルーティング処理が不要な為ルータの負荷を軽減出来る



注意点
１つ目のアクセスリスト(access-list 1 ～)
access-list 1 deny host 10.1.1.10
access-list 1 permit any

２つ目のアクセスリスト(access-list 10 ～)
access-list 10 deny host 10.1.1.10
access-list 10 permit any

例えば
fa0/0のin側にアクセスリスト１つ目を設定後、
アクセスリスト２つ目を設定すると上書きされる

(config-if)#ip access-group 1 in
(config-if)#ip access-group 10 in
2つ目のaccess-list 10のみが有効になる

ACL設定の注意事項

・ステートメントの順番

番号10、20と登録順に採番され若番から評価される

番号	送信元IP	処理
10	10.1.1.1	拒否
20	10.1.1.0/24	許可

←10.1.1.1は10.1.1.0/24に含まれるが1行目の条件に一致する為、
2行目以降は無視される

・「暗黙のdeny(拒否)」の存在

ACLの最終行には必ず暗黙のdenyが存在する為、permit行が最低一行は必要

番号	送信元IP	処理
10	10.1.1.1	拒否
20	10.1.1.0/24	許可
	暗黙のdeny	

・フィルタリングの対象パケット

ACLの対象になるパケットは、設定したルータは対象外

※ステートメント　　ACLに含まれる１行の条件文
　シーケンス番号　　シーケンス番号の数字の低い順番に評価される

ワイルドカードマスク
・32ビットの値
・8ビットずつドットで区切って10進数で表記
・「0」を指定した箇所をチェックする
・「1」を指定した箇所はチェックしない

ホストアドレスの指定
・10.1. 1 .1 0.0.0.0 = host 10.1.1.1

IPアドレス 00001010.00000001.00000001.00000001
ワイルドカードマスク 00000000.00000000.00000000.00000000

全てのアドレス
・0.0.0.0 255.255.255.255 = any

サブネットアドレスの指定(例)
・10.1.1.0/24(サブネット 255.255.255.0)

10.1.1.0 0.0.0.255

IPアドレス 00001010.00000001.00000001.00000000
ワイルドカードマスク 00000000.00000000.00000000.11111111

・10.1.1.0/26(サブネット 255.255.255.192)

10.1.1.0 0.0.0.63

IPアドレス 00001010.00000001.00000001.00000000
ワイルドカードマスク 00000000.00000000.00000000.00111111

番号付きACL
番号を指定して作成する ACL

標準ACL
　条件　送信元のIPアドレス
　番号　1～99　又は1300~1999でも指定可能

(config)#access-list <acl-number> { permit | deny | remark } <source> [<wildcard>]
acl-number･･･標準ACLの番号を1~99、1300~1999の範囲で指定。同 ACLの場合、2行目以降のステートメントも同じ番号を使用する
permlt ･･･････条件に一致した場合に許可する
deny ･･･････････条件に 致した場合に拒否する
remark･･･････ACL内にコメント文を挿入する。コメントは 100文字まで入力可能
source ･･･････送信元IPアドレスを指定
wildcard･･･････ワイルドカードマスクを指定(オプション)。省略した場合は0.0.0.0が適用される

拒否　送信元IP

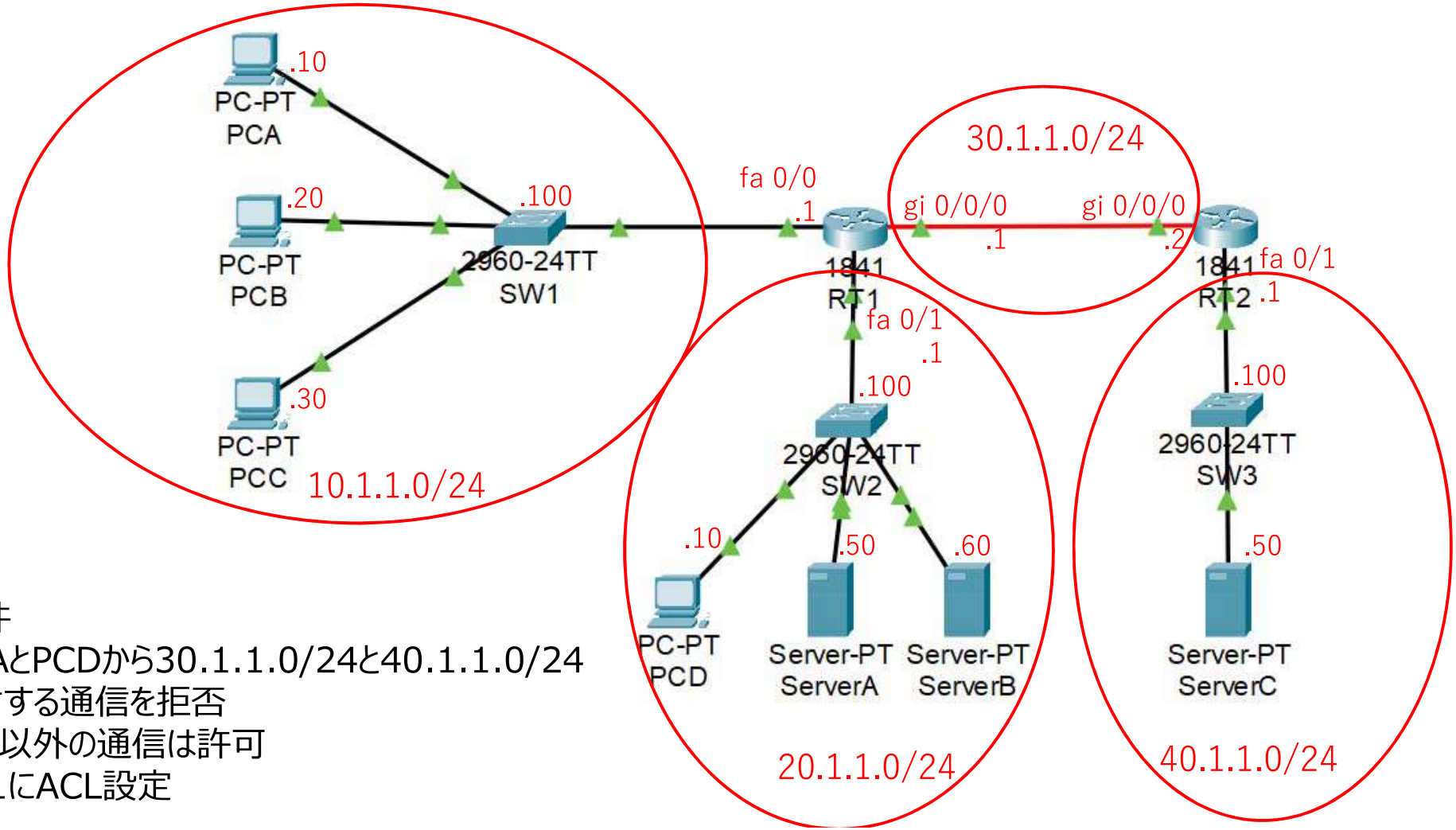
(config)#access-list 1 deny host 10.1.1.1

(config)#access-list 1 permit any

許可　送信元IP(全て)

(config-if)#ip access-group 1 in (または out)

トポロジ(4NW 標準ACL)



要件
PCAとPCDから30.1.1.0/24と40.1.1.0/24に対する通信を拒否
それ以外の通信は許可
RT1にACL設定

標準ACLを作成しRT1 gi0/0/0 のアウトバウンドにACLを適用

RT1
RT1(config)#access-list 1 deny host 10.1.1.10
RT1(config)#access-list 1 deny host 20.1.1.10
RT1(config)#access-list 1 permit any

RT1(config)#int gi 0/0/0
RT1(config-if)#ip access-group 1 out

番号付きACL
・拡張ACL
　条件　送信元、宛先IPアドレス、プロトコル、送信元、宛先ポート
　番号　100～199　又は2000~2699でも指定可能

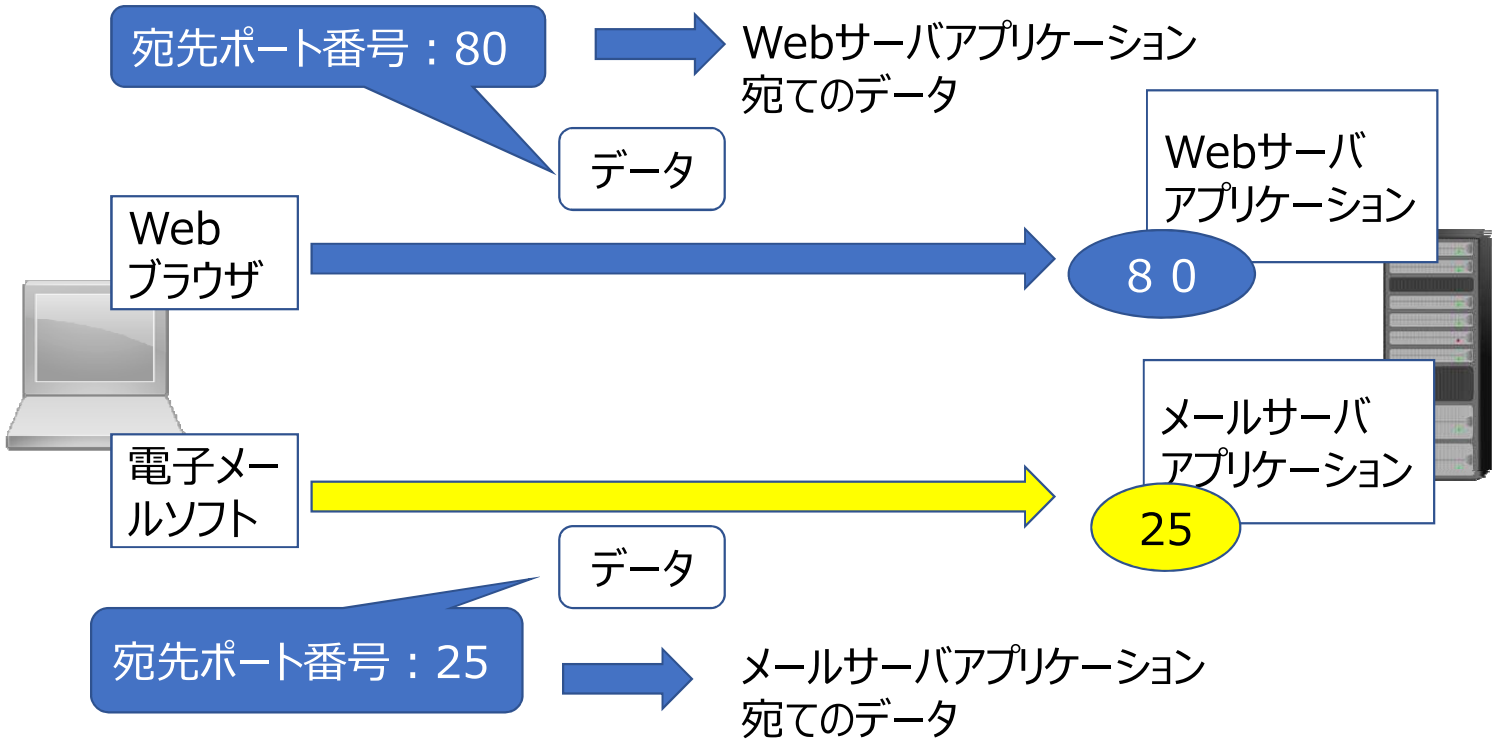
(config)#access-list <acl-number> { permit | deny | remark　} <protocol> <source> <wildcard> [<operator-port>] <destination> <wildcard> [<operator-port>] [established]

acl-number・・・・・拡張ACLの番号を100~199、2000~2699の範囲で指定。同ACLの場合、2行目以降のステートメントも同じ番号を使用する
permit・・・・・・・・・・条件に一致した場合に許可する
deny・・・・・・・・・・・・条件に一致した場合に拒否する
remark・・・・・・・・・・ACL内にコメ ント文を挿入する。コメントは 100文字まで入力が可能
protocol・・・・・・・・・プロトコル名を指定(tcp、udp、icmp、IP、ospf、eigrpなど)
source・・・・・・・・・・送信元IPアドレスを指定
wildcard・・・・・・・・・ワイルドカードマスクを指定
operator-port・・・以下の演算子 (operator)の後ろにポート番号またはアプリケーションプロトコル名を指定(オプション)
eq 等しい(equal)　　neq 等しくない(not equal)　　lt より小さい (less than)　　gt より大きい (greater than)　　range 範囲 (inclusive range)
(例)Telnetを指定する場合→ eq 23(またはeq telnet)
destination・・・・・・・・・宛先IPアドレスを指定
established・・・・・・・・・TCPのackビットが1のパケットを条件に指定(オプション)。外部からのSYN(コネクション確立要求)を拒否できる。プロトコルにtcpを指定した場合のみ使用可能

(config)#access-list 100 permit tcp host 10.1.1.1 host 20.1.1.1 eq 80

(config-if)#ip access-group 100 in (または out)

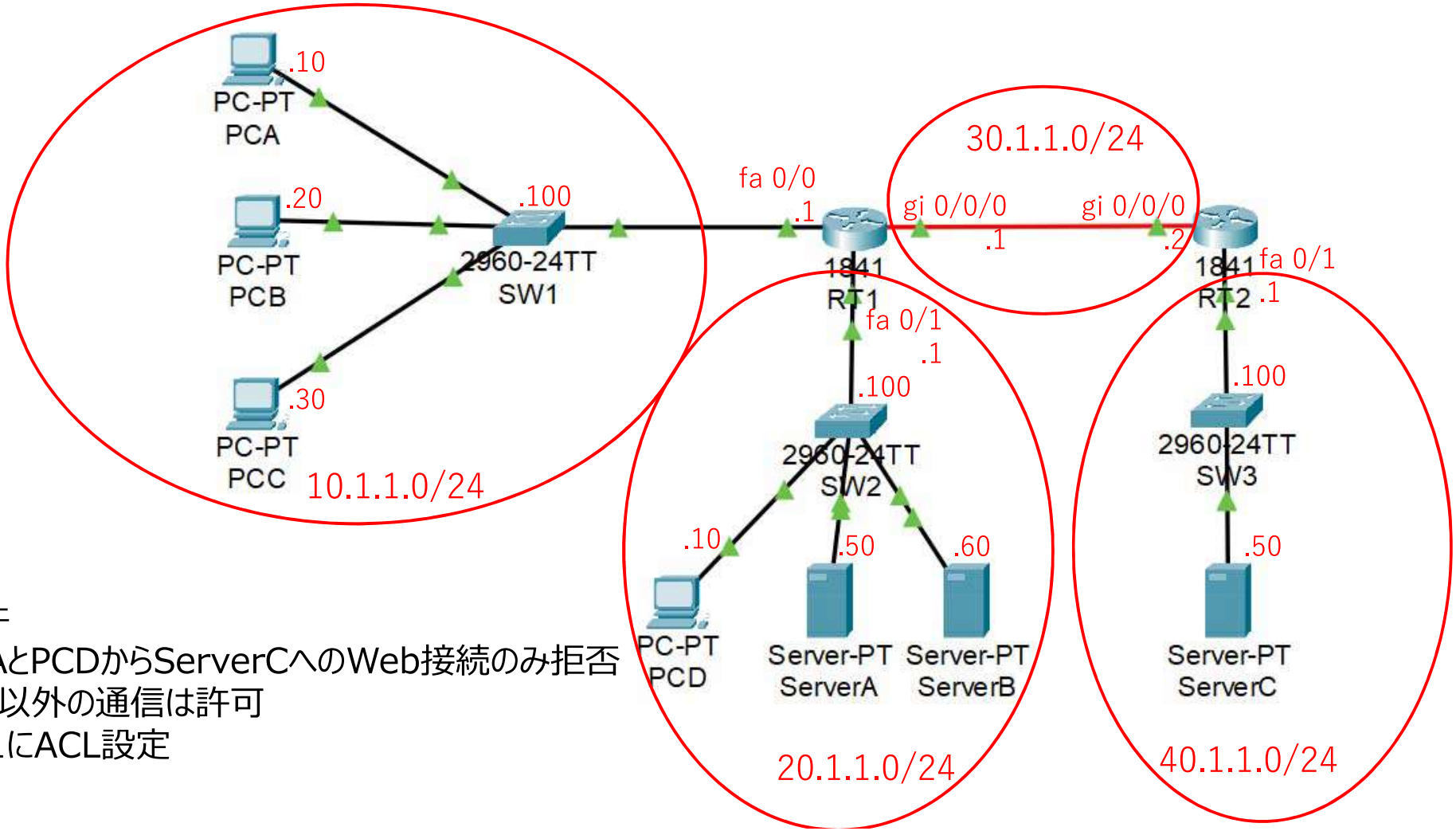
ポート番号



主なウェルノウンポート番号

プロトコル	TCP	UDP
HTTP	80	-
HTTPS	443	-
SMTP	25	-
POP3	110	-
IMAP4	143	-
DNS	53	53
FTP	20/21	-
TFTP	-	69
DHCP	-	67/68
Telnet	23	-

トポロジ(4NW 拡張ACL)



要件
PCAとPCDからServerCへのWeb接続のみ拒否
それ以外の通信は許可
RT1にACL設定

拡張ACLを作成しRT1 gi0/0/0 のアウトバウンドにACLを適用

RT1
RT1(config)#access-list 100 deny tcp host 10.1.1.10 host 40.1.1.50 eq 80
RT1(config)#access-list 100 deny tcp host 20.1.1.10 host 40.1.1.50 eq 80
RT1(config)#access-list 100 permit IP any any

RT1(config)#int gi 0/0/0
RT1(config-if)#ip access-group 100 out

名前付きACL
例えばFTPトラフィックを制御する場合はftp-filterとするなど、目的に合った名前でACLを作成すると管理しやすくなる
標準(Standard) / 拡張(Extended)

名前付き標準ACLの作成
(config)#IP access-list standard <acl-name>
(config-std-nacl)#[<sequence-number>] { permit | deny | remark } <source> [<wildcard>]
[]は省略可能
・ acl-name・・・・・・・・・・ACLの名前を定義。標準ACLの範囲で番号を指定する事も可能
・ sequence-number・・・シーケンス番号(行番号)を 1~2147483647の範囲で指定。
省略すると1行目のステートメントは10番、以降は10ずつ増加(オプション)
※その他の引数は番号付き標準ACLと同様

要件:192.168.1.0/24から外部ネットワークへのアクセスのみ拒否し、
それ以外のトラフィックはすべて許可

(config)#ip access-list standard ftp-filter
(config-std-nacl)#deny 192.168.1.0 0.0.0.255
(config-std-nacl)#permit 0.0.0.0 255.255.255.255
(config-std-nacl)#exit
(config)#int fa0/0
(config-if)#ip access-group ftp-filter out

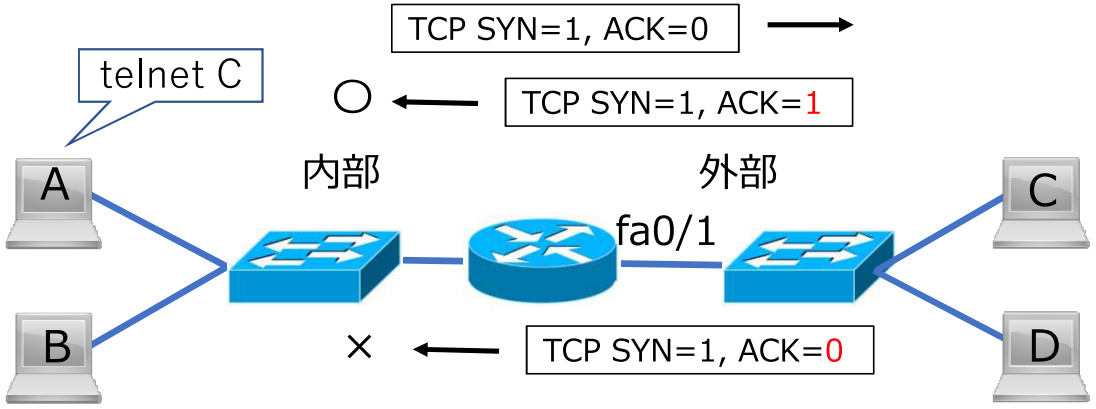
名前付き拡張ACLの作成
(config)#ip access-list extended <acl-name>
(config-ext-nacl)#[<sequence-number>] {permit | deny | remark } <protocol> <source> <wildcard> [<operator-port>] <destination> 〈wildcard> [<operator-port>] [established]

established
「内部から発した通信を許可し、外部からの通信を拒否したい」ときに使用

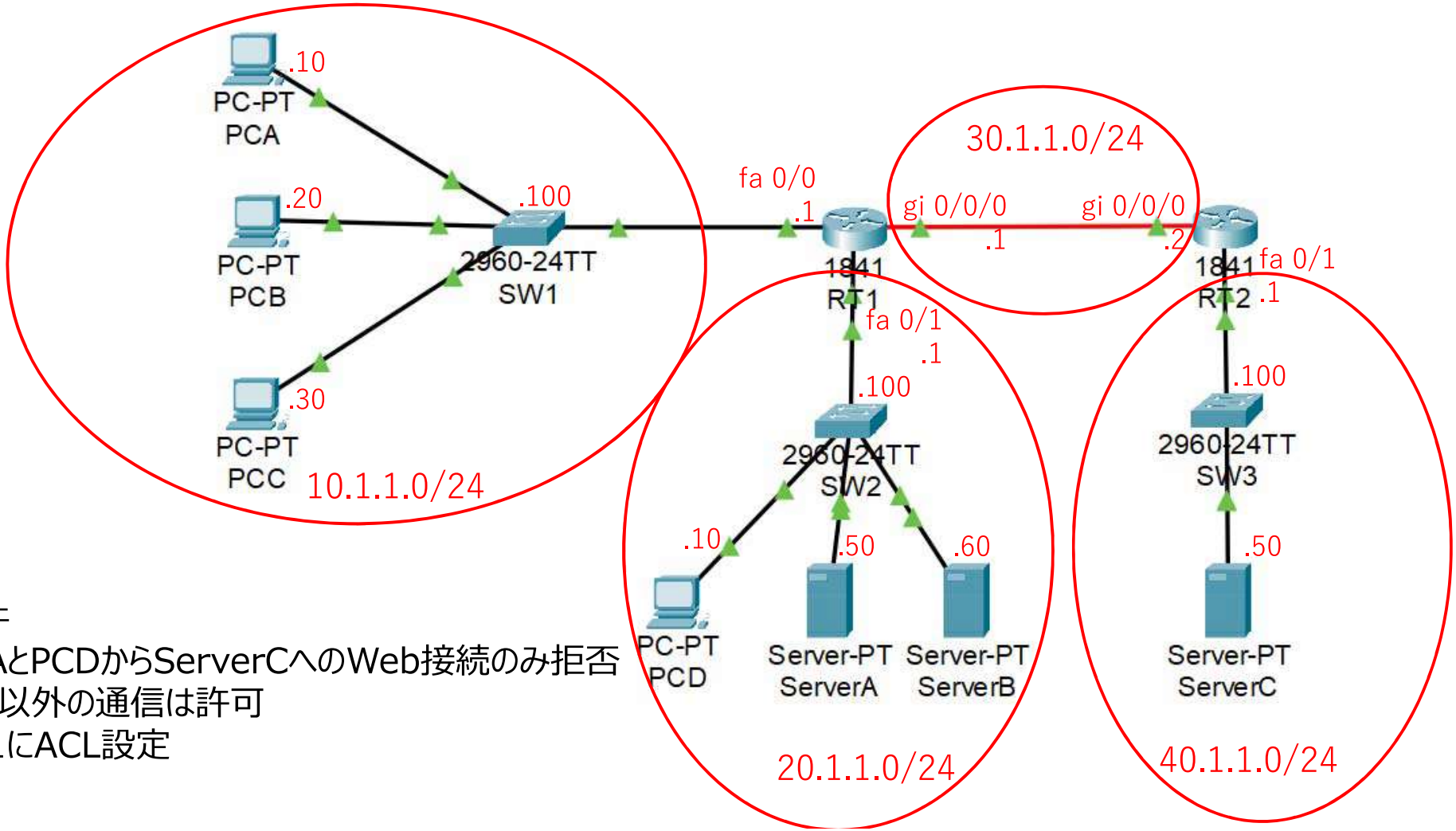
TCPのACKビットが1のパケットを条件に指定(オプション)
外部からのSYN(コネクション確立要求)を拒否できる
プロトコルにtcpを指定した場合のみ使用可能

要件:内部ネットワークから発信したTCPセッションを許可し、
外務ネットワークから発信されたTCPセッションを拒否
外部ネットワークからのICMPエコー応答を許可
外部ネットワークからのICMP宛先到達不能を許可
外部ネットワークからのICMP時間超過を許可
外部ネットワークからのそれ以外のパケットはすべて拒否

```
(config)#ip access-list extended lan-block
(config-ext-nacl)#permit tcp any any establihed
(config-ext-nacl)#permit icmp any any echo-reply
(config-ext-nacl)#permit icmp any any unreachable
(config-ext-nacl)#permit icmp any any time-exceeded
(config-ext-nacl)#exit
(config)#int fa 0/1
(config-if)#ip access-group lan-block in
```



トポロジ(4NW 名前付き拡張ACL)



要件
PCAとPCDからServerCへのWeb接続のみ拒否
それ以外の通信は許可
RT1にACL設定

名前付き拡張ACLを作成しRT1 gi0/0/0 のアウトバウンドにACLを適用

RT1
(config)#ip access-list extended web-block
(config-ext-nacl)#deny tcp host 10.1.1.10 host 40.1.1.50 eq 80
(config-ext-nacl)#deny tcp host 20.1.1.10 host 40.1.1.50 eq 80
(config-ext-nacl)#permit IP any any
(config-ext-nacl)#exit

(config)#int gi 0/0/0
(config-if)#ip access-group web-block out

参考番号付き拡張ACLの適用
RT1(config)#access-list 100 deny tcp host 10.1.1.10 host 40.1.1.50 eq 80
RT1(config)#access-list 100 deny tcp host 20.1.1.10 host 40.1.1.50 eq 80
RT1(config)#access-list 100 permit IP any any

RT1(config)#int gi 0/0/0
RT1(config-if)#ip access-group 100 out

ACLの用途
ACLはトラフィックを分類して区別する事が出来る
分類する事によってトラフィックを「特別な処理」の対象に割り当てる事が出来る

たとえば、次のような用途でACLは利用
・NAT(Network Address Translation)
内部ネットワークかうインターネット上の宛先へパケットを転送する際、NAT機能によってプライベートIPアドレスをグローバルIPアドレスに変換する対象パケットにするかどうかをACLで分類

・ルートフィルタリング
ルーテイングプロトコルによって経路情報をアドバタイズする際に、特定の経路情報を通知しないようにACLで指定(分類)

・VPN (Virtual Private Network)
パケッ トをVPNによって転送するかどうかをACLで分類

・ルート再配布
ルート再配布の際に経路情報を再配布の対象にするかどうかをACLで分類
このほかにも、ACLの条件を利用して様々な処理を制御する事が出来る
また、ACLはルータやスイッチなど幅広い製品でサポート