SÉCURITÉ.ORG

Internet Service Provider Infrastructure Security

Nicolas FISCHBACH

Senior Manager, IP Engineering/Security - COLT Telecom

nico@securite.org - http://www.securite.org/nico/

version 1.0

COLT

we make business straight.forward

Agenda

- » **Router Security**
  - > Router security basics
- » **Infrastructure Security**
  - > Filtering, BGP/DNS
  - > Forensics
- » **Distributed Denial of Service**
  - > Trends in attacks, worms and botnets
  - > Detection and mitigation
- » **Other recent and new risks**
  - > IPv6, MPLS, Lawful Intercept, SPAM, etc.
- » **Conclusion**

Router Security

» **Hardware**

> Depending on the model/series (at least)

- mother board
- CPU (RISC - MIPS or Motorola)
- memory
- bus
- I/O interfaces

> Becomes much more complex (GSR for example)

- distribute tasks (CPU takes only care of basic “running the system” tasks and not routing/forwarding)
- Line Card (own CPU), Engines, etc.
- ASICs

Router Security

» **Memory**

- > Flash (non volatile)
  - contains the (compressed) IOS image and other files
- > DRAM/SRAM (volatile)
  - contains the running IOS
  - store the routing table(s), statistics, local logs, etc.
  - divided into regions (processor, I/O, I/O 2).
- > NVRAM (non volatile)
  - contains the startup configuration (*startup-config*)
  - *boot config <file system><config>* configures an alternative location
- > BootROM
  - contains the ROMMON code (POST, IOS loading, etc.)

Router Security

» **IOS**

> Proprietary, closed source OS running on RISC CPUs

> Closed source, closer to a “port” than a “fork” from (BSD) Unix (zlib, ssh, SNMP bugs, etc.)

> ELF 32-bit MSB executable, statically linked, stripped

> IPCs for communications between the RP (Route Processor and the LCs (Line Cards) on the GSR series

“Inside Cisco IOS software architecture” - Cisco Press :
- “In general, the IOS design emphasizes speed at the expense of extra fault protection”
- “To minimize overhead, IOS does not employ virtual memory protection between processes”
- “Everything, including the kernel, runs in user mode on the CPU and has full access to system resources”

Router Security

» **Cisco IOS rootkit / BoF / FS : open questions / issues**

> No (known) local tools/command to interact and “play” with the kernel, memory, processes, etc.

  - What is possible with gdb (gdb {kernel¦pid pid-num}) ?
  - Is the ROMMON a good starting point (local gdb) ?

> What can be done in enable engineer mode (Catalyst) ?

> Is it possible to upload a modified IOS image and start it without a reboot ?

> A lot of different images exists and are in use - what kind of tool would be needed ?

> What will happen with IOS-NG (support for loadable modules) ?

Router Security

» **Before going live**

> Turn off all the unneeded services

- See “Protecting your IP network infrastructure”, slides 44+
- New features in 12.3
  - auto-secure script
  - local accounting in XML format

> Lots of data are volatile: log/poll as much as you can (but keep CPU and/or memory impact in mind)

- (authenticated) NTP sync.
- run syslog (local, size limited buffer)
- log events generated by services (routing protocols for ex.)
- SNMP traps/poll
- AAA logs and events
- ../..

Router Security

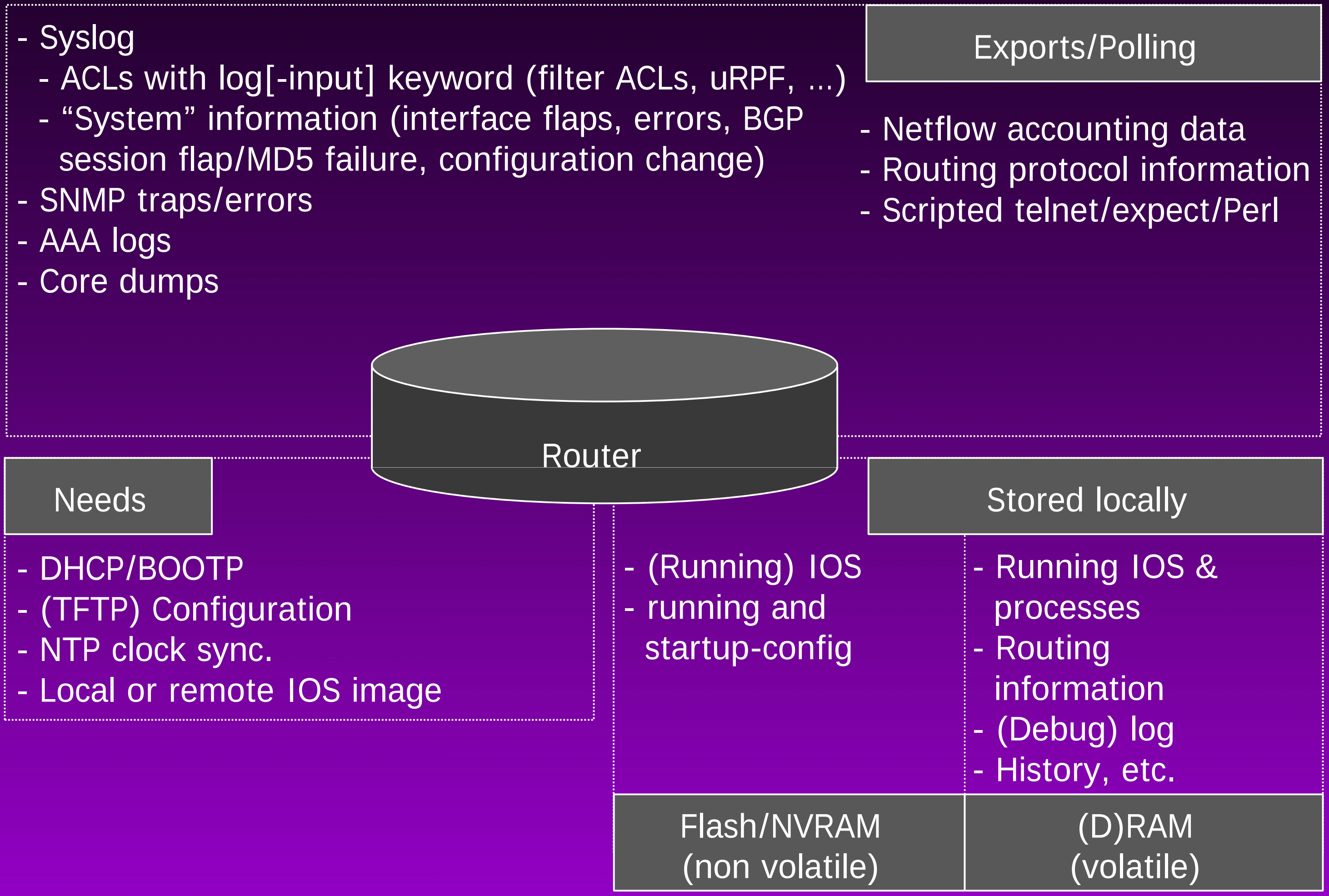
» **Before going live (cont.)**

> Lots of data are volatile: log/poll as much as you can (but keep CPU and/or memory impact in mind)

- Netflow accounting flows
- core dump (automatic upload)
- ACLs (filtering and application/service access control)
- *config-register* (Configuration Register) - 0x2102
- scheduler tuning
- *debug sanity* (checks on malloc/free, performance impact)

Router Security

» Available data and elements



- Syslog
  - ACLs with log[-input] keyword (filter ACLs, uRPF, ...)
  - “System” information (interface flaps, errors, BGP session flap/MD5 failure, configuration change)
- SNMP traps/errors
- AAA logs
- Core dumps

Exports/Polling

- Netflow accounting data
- Routing protocol information
- Scripted telnet/expect/Perl

Router

Needs

- DHCP/BOOTP
- (TFTP) Configuration
- NTP clock sync.
- Local or remote IOS image

Stored locally

- (Running) IOS
- running and startup-config

- Running IOS & processes
- Routing information
- (Debug) log
- History, etc.

Flash/NVRAM (non volatile)

(D)RAM (volatile)

Router Security

» **Four steps to build a tripwire-like for IOS/CatOS**

> 1. Store your routers and switches configurations in a central (trusted and secure) repository (CVS for example)

> 2. Get the configuration from the device (scripted telnet, Perl, expect, tftp, scp, etc.) or have the device send you the configuration (needs a RW SNMP access - not recommended)

```
snmpset -c <community> <router’s IP> .1.3.6.1.4.1.9.2.1.55.<TFTP server’s IP> s <file>
```

> 3. Check : automatically (cron/at job), when you see “configured by <xyz>” or a router boot in the logfile or when you get the “configuration changed” SNMP trap

> 4. Diff the configuration with your own script or use tools like CVS, Rancid, CW, etc.

Router Security

» Limitations and details

- > You still have to trust the running IOS/CatOS (no Cisco “rootkit” yet) and your network (MITM attacks)
- > The configuration is transmitted in clear text over the network (unless you use scp or IPsec to encrypt the traffic)
- > Do not forget that there are two “files”: startup-config and running-config
- > Do the same for the IOS/CatOS images
- > Cisco MIBs : CISCO-CONFIG*

Router Security

» **Decisions**

> Depending on your network architecture: effect on the network availability

- no routing/forwarding
- cold/hot spare (flash, NPE/RP, LC, etc.)

> How to connect ?

- Telnet/SSH or via the console or serial port ?

> What needs to be done before and after reboot

- local logs and (enable) commands to use
- which configuration register to use (*config-register*) ?

> If you can't connect/change to *enable* mode on the router ?

- password reset/recovery
- *nmap*, *snmpwalk*, etc.
- network environment

Router Security

» Commands to use

- > Make sure you save all the commands and output !
- > Avoid entering the configuration mode
- > “enable”/”user” EXEC mode ?

Configuration and users

show clock detail
show version
show running-config
show startup-config
show reload
show users/who

Local logs, process and memory

show log/debug
show stack : stack state
show context : stack information
show tech-support : incomplete
show processes {cpu, memory}
content of *bootflash:crashinfo*

Network informations

show ip route
show ip ospf {summary, neighbors, etc)
show ip bgp summary
show cdp neighbors : Cisco Discovery Protocol
show ip arp
show {ip} interfaces
show tcp brief all
show ip sockets
show ip nat translations verbose
show ip cache flow : Netflow
show ip cef : Cisco Express Forwarding
show snmp {user, group, sessions}

File systems

show file descriptors: lsof like
show file information <url>: file like

Router Security

- » ***debug* mode**
- » **Flash memory**
  - > Details on the content (files, state, type, CRC, etc)
    - - *show <file system>*
  - > Ciscoflash: ftp://ftp.bbc.co.uk/pub/ciscoflash/
- » **DRAM / SRAM**
  - > Informations on memory regions
    - - *show buffers*
    - - *show memory*
    - - *show region*
- » **NVRAM**
  - > Information about the startup configuration/mode
    - - *show bootvar*

Router Security

» **Environment**

> Application logs
>
> - syslog, TACACS, NMS, etc.
>
> Side effect on network traffic and the infrastructure ?
>
> Network traces
>
> - IDS
> - Mirror (SPAN) port on a switch (depending on the architecture) or RTE on a router
> - Netflow exports
> - In-line devices/taps

» **General recommendations**

> Document and date every single step
>
> Use out-of-band communications as much as possible

Router Security

» **Router Security 101**

- > Good infrastructure security starts with good router security
- > Packet forwarding vs “received” packets performance
- > Like on any system:
  - Use VTY (virtual TTY) ACLs, avoid passwords like “c”, “e”, “cisco”, “c1sc0” and use an AAA system like TACACS+
  - Avoid shared accounts and use privilege levels/restrict commands
  - Secure in/out-of-band management
  - Turn off unneeded services, restrict SNMPd, configure management ACLs
  - Activate logging (but not too much!)
  - Configuration and ROMMON/IOS images integrity
  - Make your router “forensics ready” (lots of “volatile” data)

Router Security

- **Router Security 101**
  - Your biggest security risk ?
    - The Customer Diagnostic/NOC guy leaking configurations to customers that include shared/common passwords and communities, the management ACLs, TACACS+ server IPs and shared keys, etc.
    - Think filtering scripts/peer approval
  - Like with any program or application: don’t trust client input
    - What could happen if the customer unplugs your managed router and plugs his own router (management ACLs, filtering, etc) ?

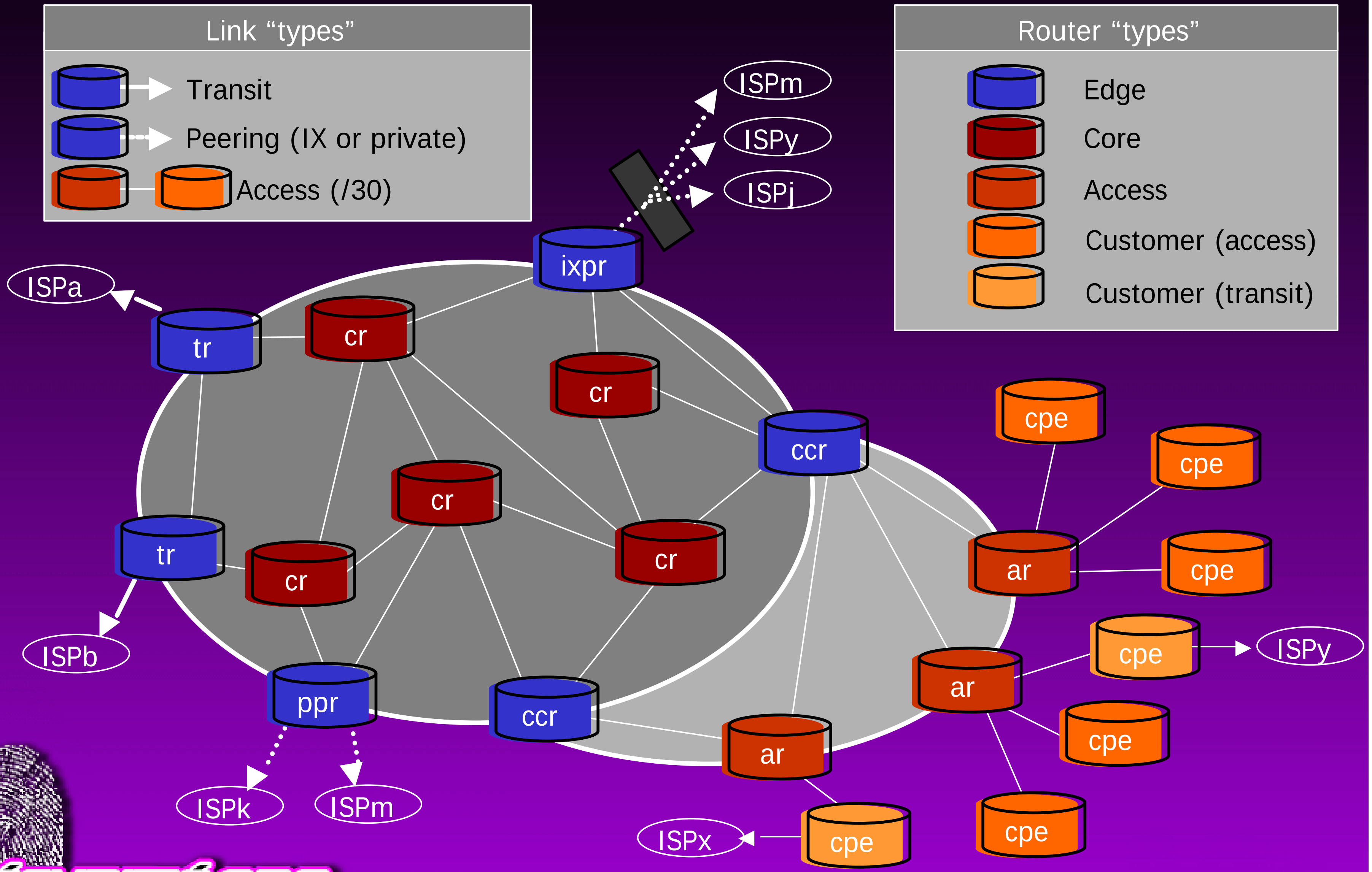
Infrastructure Security

» **Infrastructure Security**

- > The Internet is considered a “critical infrastructure”
- > Filtering routing information and filtering traffic (IP layer) are complementary
- > BGP and DNS are the core protocols
- > Your backbone: large firewall or transit network ?
- > Data-center vs core infrastructure based detection
  - - Data-center: in-line (“complete packet”)
  - - Infrastructure/distributed: Netflow (“header only”)
  - - Find the right mix of both
    - . Scalability
    - . CAPEX
    - . Sampled Netflow (high probability of missing single packets) vs one in-line device (mirrored traffic) per larger POP

FrNOG 4

Infrastructure Security



Link “types”

- Transit
- Peering (IX or private)
- Access (/30)

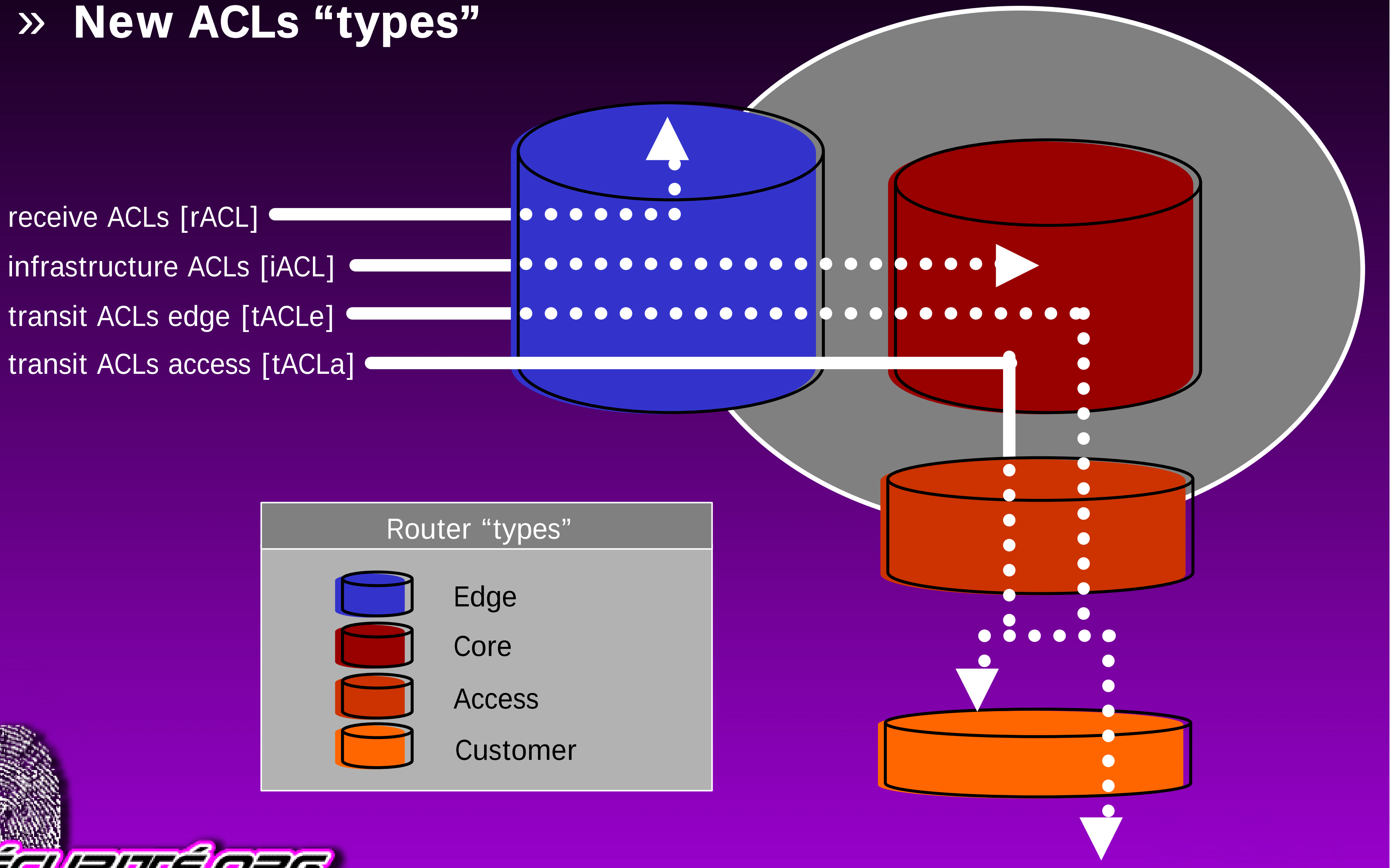
Router “types”

- Edge
- Core
- Access
- Customer (access)
- Customer (transit)

SÉCURITÉ.ORG

Infrastructure Security

» New ACLs “types”


receive ACLs [rACL]
infrastructure ACLs [iACL]
transit ACLs edge [tACLe]
transit ACLs access [tACLa]
Router “types”
Edge
Core
Access
Customer

Infrastructure Security

- **New ACLs "types"**
  - iACLs: why should anybody with Internet connectivity be able to "talk" to your network core ? (traffic directed at the infrastructure)
    - you need a structured address plan
  - rACLs: helps to protect the Route Processor (traffic directed at the router)
  - tACLs: enables filtering on the forwarding path (traffic "transiting" your network)
  - Keep them short and generic, avoid exceptions
  - "Default permit" or "default deny" ?

Infrastructure Security

» **New ACLs "types"**

> Combine them with anti-spoofing ACLs/uRPF at the edge

> Don't forget management traffic (telnet/SSH, SNMP, TFTP, syslog, AAA, etc) and routing protocols

> What to do with ping and traceroute (ICMP/UDP): incoming and outgoing (for troubleshooting)

» **Other types of "filtering"**

> Re-coloring (QoS): enforce it at your AS boundaries

> Rate-limiting: what to throttle and what does it break ?

> Other options to protect the router

- rate-limit the traffic to the RP (data punt/slow path)
- Avoid "administrative traffic generating options" (like ACLs with logs)
- IP options, ICMP, mcast "filtering", etc.

Infrastructure Security

» **ACLs (Access Control Lists)**

- > Always (try to) use compiled ACLs: avoid log[-input], source port, output ACLs, etc.
- > Where to filter: edge, core, transit, peerings ?
- > What to filter: protocols, src/dst IP/ports, header, payload ?
- > Who should filter: tier1, tier 2/3 providers (with broadband home users), enterprise (FWs) ?
- > In which direction: to and/or from the end-users (ie. protect the Internet from the users and/or vice-versa) ?
- > Depending on the hardware and software capabilities: micro-code/IOS and engines (-: 0, 1, 4; +: 2; ++: 3)
- > Scalability of the solution (no easy way to maintain distributed ACLs policies)
- > How long should you keep these filters in place ?

Infrastructure Security

» **uRPF (unicast Reverse Path Forwarding)**

- > Strict uRPF for single-homed customers (route to source IP points back to the ingress interface)
- > Loose uRPF for multi-homed customers (route/network prefix present in the routing table)
- > Loose uRPF doesn’t protect from customer spoofing
- > Adapt strict/loose policy depending on your customers’ setup
- > Statistics prove that uRPF is not really deployed (nor loose, nor strict)

Infrastructure Security

» **Other ("edge"-only) features**

- > NBAR (Network Based Application Recognition)
  - - Used with custom Cisco PDLMs (Packet Description Language Module) to identify P2P traffic in quite some university networks
- > TCP Intercept
  - - Usually done by the enterprise FW
- > What else do you want you router to do for you today ? ;-)

Infrastructure Security

» **BGP (Border Gateway Protocol)**

> Not as easy as many think (and say) to hijack BGP sessions!

> BGP flaps (dampening) and configuration mistakes

> Trivial passwords and no VTY ACL on a BGP speaking router: cool “warez” for underground/SPAM communities (like eBay accounts or valid CC numbers) and honeyrouters

> Filtering:

- Default-free routing in the core (to avoid the magnet effect)
- Apply the same strict policy to transit/peerings than to customers (AS_path, prefixes, max-pref, RIR allocations, etc)
- Martian/Bogons/RFC1918/RFC3330 (static or route-server ?)
- ISPs stopping to announce/route/filter the AR<->CPE /30
- Account for BGP sessions (especially in full-mesh deployments, on RRs and on peering routers) and use md5

Infrastructure Security

- **BGP (Border Gateway Protocol)**
  - Origin-AS/prefix relation is never verified
  - AS_path to key locations (especially DNS root/gtld servers)
    - Secure BGP
      - RIRs to run PKIs and act as CAs
      - Verify "ownership" (Origin-AS/prefix)
      - Signed BGP Update message
    - SoBGP
      - Distributed Origin-AS/prefix check
      - New "BGP Security" message
- **IGP (Internal Gateway Protocol) and Layer 2**
  - Scope is much more limited, but don't forget to secure it (OSPF, IS-IS, etc): filtering and md5
  - Layer 2: CDP, xTP protocols, VLANs, etc.

Infrastructure Security

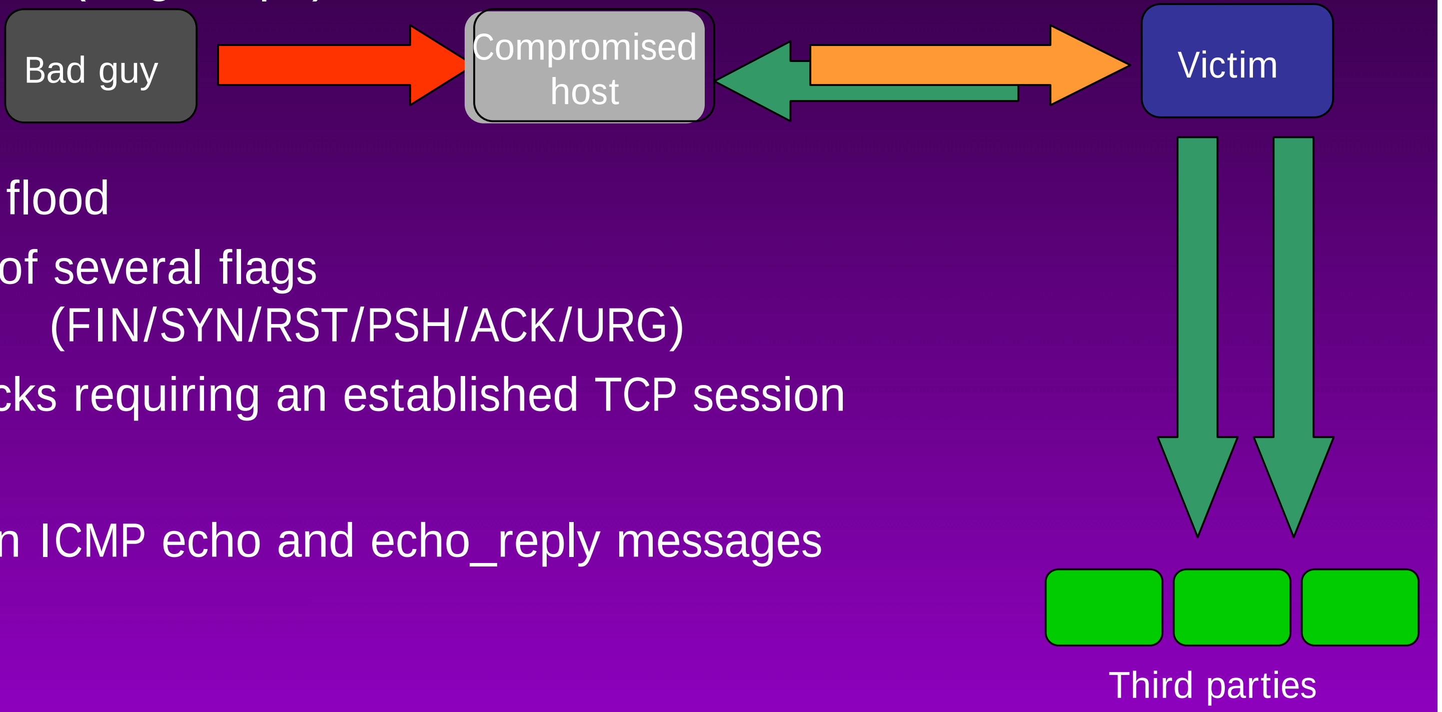
- » **DNS (Domain Name System)**
  - > Quite a few attacks recently
  - > DNS “abuse” due to bad network/system setups and broken clients: AS112 project (distributed servers to answer negative RFC1918 PTR queries)
  - > IP anycast helps but makes debugging more difficult (which server is actually producing the error ?)
  - > Key to watch Origin-AS and AS_path from/to root and gtld DNS servers
- » **Is BGP/DNS “hijacking” a real threat ?**

Distributed Denial of Service

» Basic attack

> Some (old) names :

- (win)nuke, ping of death, land, teardrop, jolt, pepsi, bo(i)nk, nestea(2), naptha , 3wahas, stream, fraggle, or a mix of some attacks (targa/rape)



> TCP

- SYN flood
- Use of several flags (FIN/SYN/RST/PSH/ACK/URG)
- Attacks requiring an established TCP session

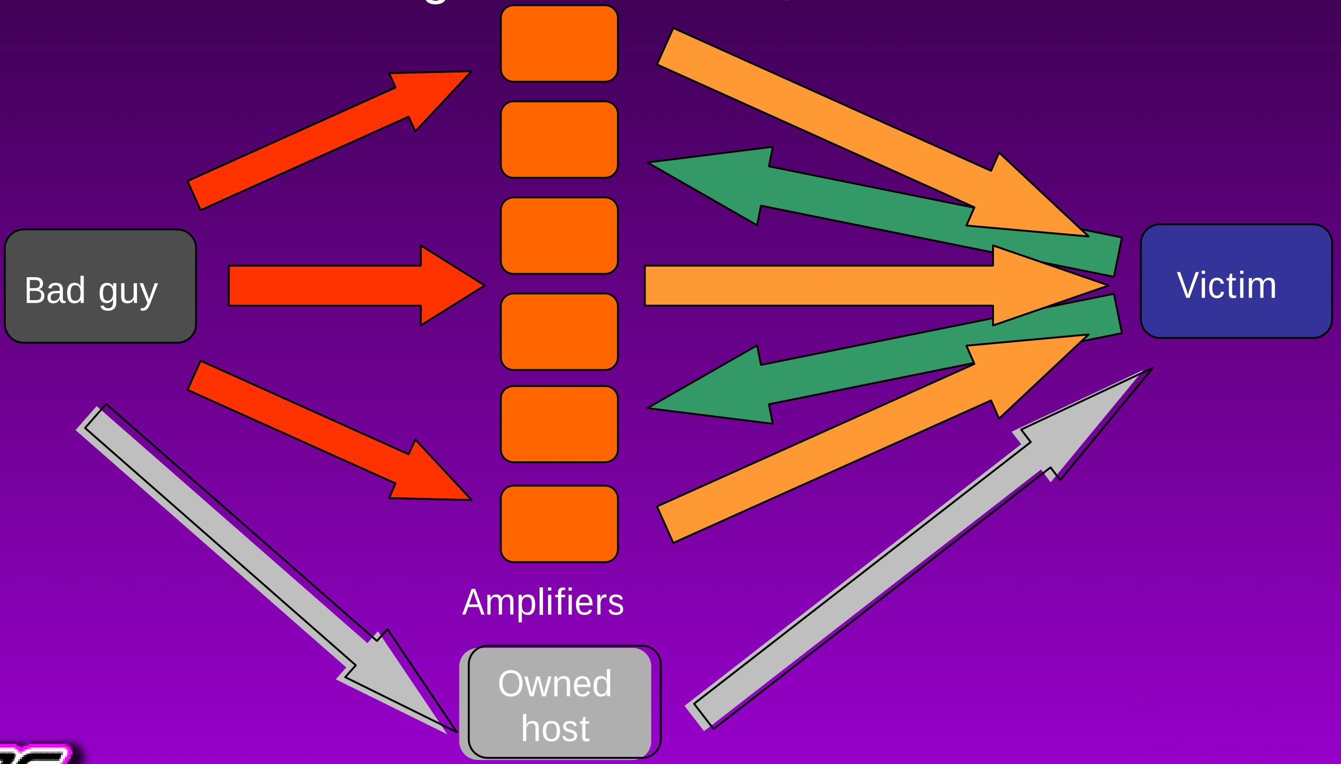
> ICMP

- Often ICMP echo and echo_reply messages

> UDP

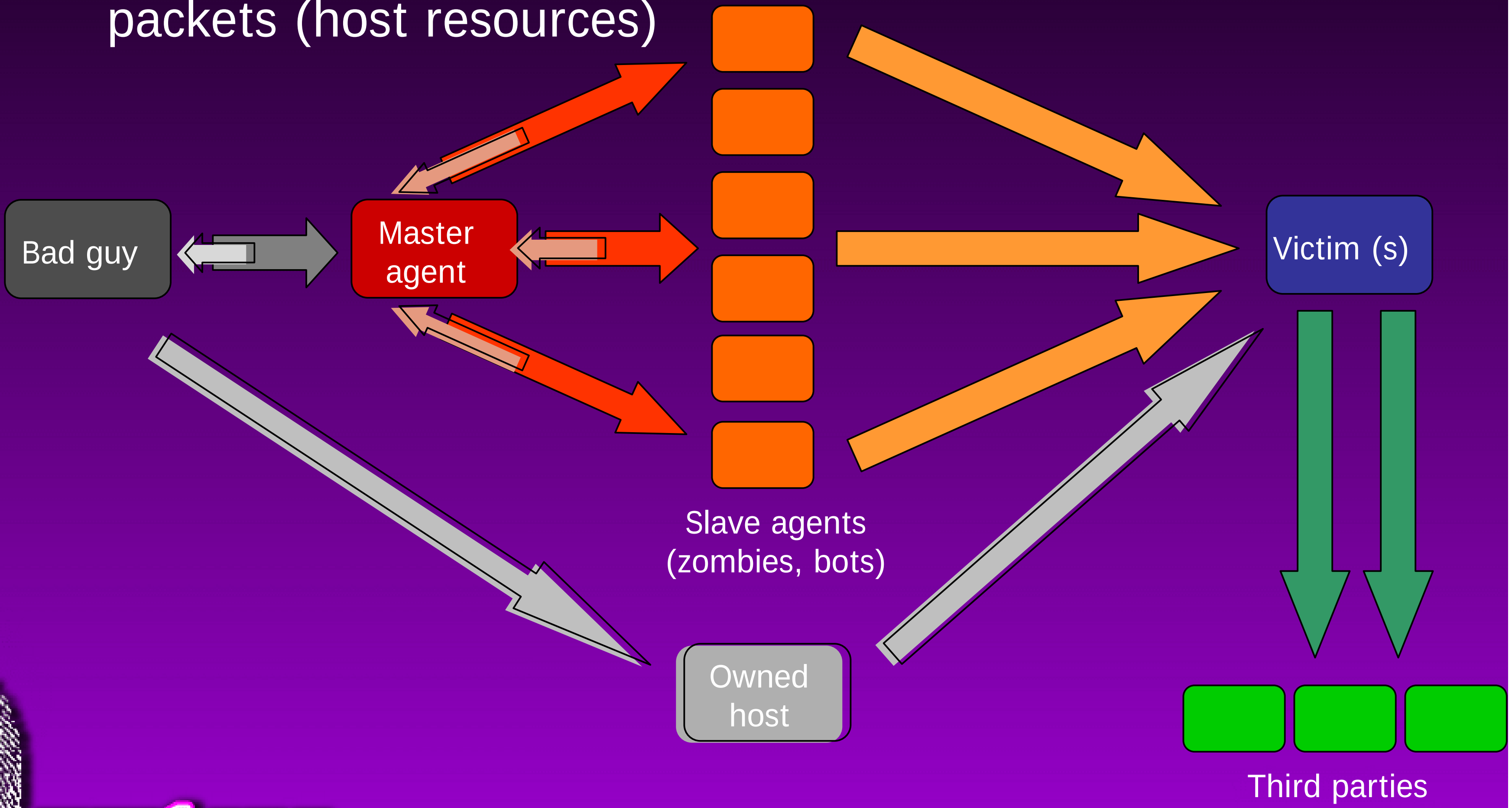
Distributed Denial of Service

- » **Amplified or reflectors based attacks**
  - > Basic attack, but amplified (factor 10-1000:1) and/or using reflectors (usually a 1:1 ratio) :
    - smurf, P2P clients/servers, DNS servers, broken TCP implementations with guessable ISNs, etc.



Distributed Denial of Service

- » **Distributed attack**
  - > Usually only one target : large packets (bandwidth), small packets (host resources)



Distributed Denial of Service

- **Slave agents**
  - « Modified » servers, services and also network equipment (ie. routers)
  - Compromised servers run a (D)DoS agent :
    - old: Trinoo, TFN{(2,3)k}, omega, Stacheldrat*, Carko, Trinity, etc.
    - sdbot, agobot, gaobot, spybot and variants
    - Trojan horse and worms
  - P2P (peer-to-peer) tools
- **Agents are distributed**
  - On the same network : school, company, ISP, cable/xDSL « area »
  - Same country or continent
  - Same « type » of network : IPv6 island, mbone, Internet2
  - Completely distributed over the Internet

Distributed Denial of Service

» Agents deployment and communications

- > « By hand »
- > Automated script (downloading data from a central server over HTTP/FTP/DCC/etc)
- > DDoS agents « deployed » using a worm or a virus and hidden using a *{tool,root}kit* (adore, t0rn, etc) :
  - Makes it easy and quick to collect and acquire a lot of systems
  - First sign of a « soon to be launched » attack
  - VBS/*, Win32/*, Code*, Nimda, 1i0n/ramen, slapper, etc.
  - (Bio)diversity helps to reduce exposure to a worm, but makes the IS more complex
- > Warez FTP servers
- > Fake update for a well known application
- > IRC, P2P tools, instant messaging, etc.

Distributed Denial of Service

» **Trends in DDoS**

- > Yesterday: bandwidth abuse, exploiting bugs, TCP SYN, UDP and ICMP floods (amplifiers)
- > Today: DDoS extortion
  - PPS (packet-per-second), against the SP infrastructure, non-spoofed sources (who cares if you have 150k+ bots anyway) and reflectors
  - Short lived route announcements (for SPAM usually)
- > Tomorrow:
  - QoS/"extended header"
  - CPU (crypto intensive tasks like IPsec/SSL/TLS/etc)
  - Protocol complexity and other attacks hidden/mixed with or even part of normal traffic where complete state information/traffic needs to be tracked ?
  - Non-cached items in distributed content networks

Distributed Denial of Service

- **Trends in worms**
  - The “worms of the summer”, bots and botnets and their effect on routing stability
  - “Old” worms still very active: patch management ?
  - What if the guys who wrote recent worms had a clue or different objectives ?
    - Worm “engines” becoming better, more distributed payload
    - Worms == SPAM (i.e. going commercial) ?
  - Which policies do SPs apply: leave everything open until it hurts the infrastructure or block for days on early warning ?
  - Can we win the race (analyze and mitigate in <1h) ?
  - After “everything on top of IP” the trend is “everything on top of HTTP[s]” (ie. circumventing firewalls 101): what if the next one is going over 80/tcp ? ;-)

Distributed Denial of Service

» **DDoS Detection**

- > ACLs, queue counters, NMS (CPU, interface counters, etc)
- > Netflow and dark IP space/bogons/backscatter monitoring
- > “Honeybot” approach
  - - Watch IRC/P2P/etc based communications
  - - Run bots in “safe mode”
- > Customers ;-)
- > Backscatter data

» **DDoS Mitigation**

- > ACLs and CAR (rate-limit)
- > null0 routing (blackholing), (anycast) sinkhole, shunt, traffic rerouting and “cleaning”
- > Propagated blackholing (special community)

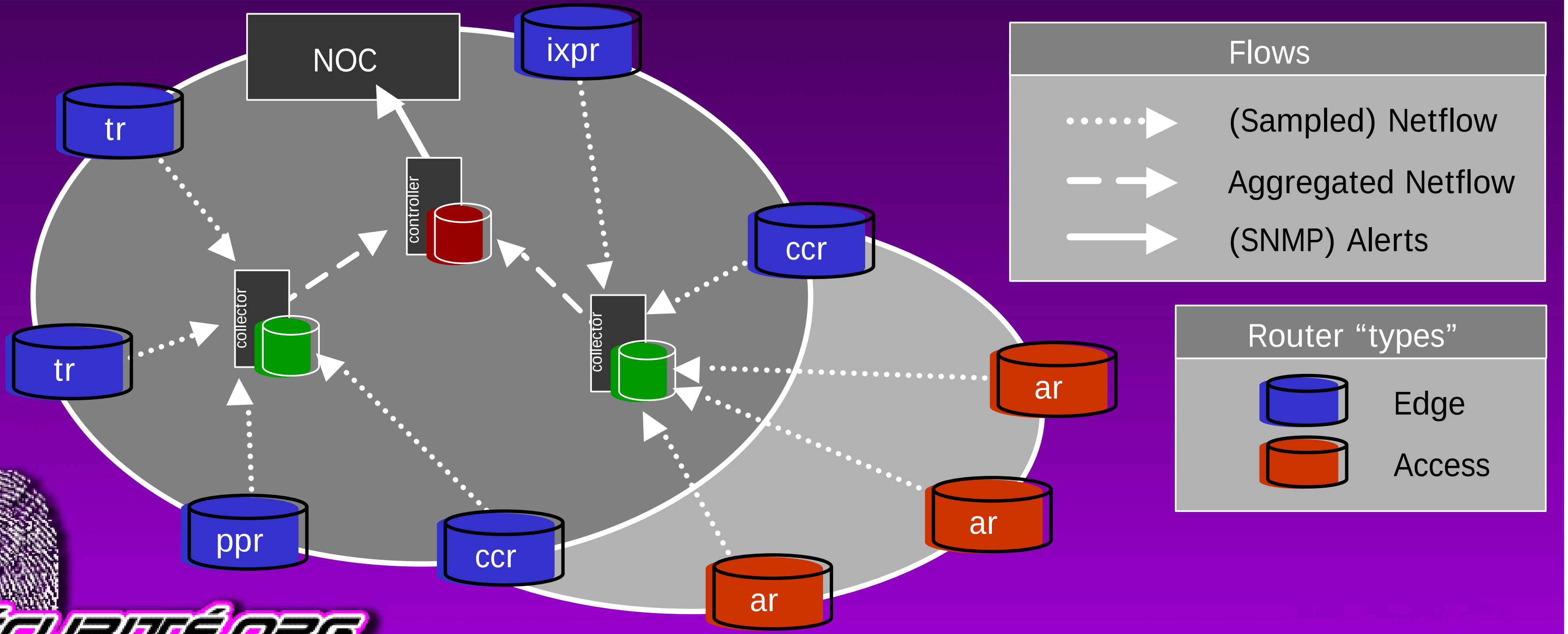
Distributed Denial of Service

» **Netflow based detection**

> Flow (src/dst IP/port, protocol, ToS, interface - no payload)

> Usual traffic distribution (90% TCP, 8% UDP, <1% ICMP/GRE/IPsec/others - 50% of small packets)

> Needs as much fine tuning as an IDS



Distributed Denial of Service

» **Forensics: BGP, Netflow (and ACL logs)**

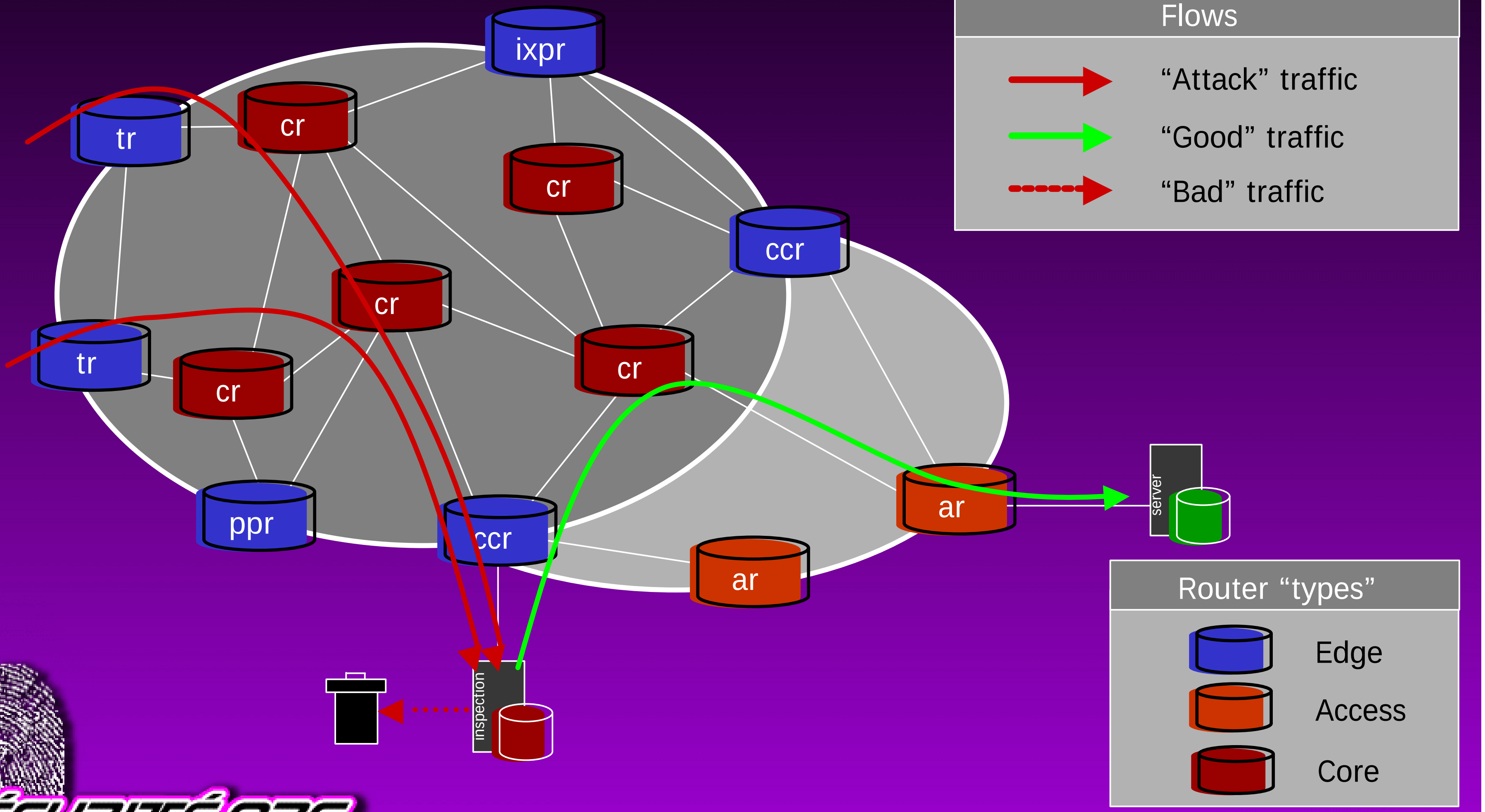
- > Hop-by-hop DDoS attack tracing using ACLs or ip source-tracker isn't very effective
- > BGP Update messages and (sampled Netflow) accounting will be part of the next-generation high-bandwidth IDSes and a must for historical data: Netflow for the more high level view (ie. the flow) and traffic dumps for the low level view (ie. the actual data)
- > Distributed Route Collectors give a much better view
- > Putting these bits together create a good anomaly detection system and good source for historical data (next to enabling you to do better traffic management ;-)

Distributed Denial of Service

» **Traffic diversion (and inspection/cleaning)**
  - > The alternative to strict filtering (which usually means the attacker won) ?
  - > Required when layer3+ and stateful information is needed
  - > BGP and/or Policy Based Routing (PBR) as the triggering mechanism(s)
  - > Tunnels: MPLS, GRE, L2TPv3, IPsec, etc.
  - > Such “cleaning centers” should be distributed across your network (large POPs, known attack entry points, etc)
  - > Same concept can be applied to honeynets (distributed honeynets/honeyfarms)
  - > Issues: inter-city capacity, eng0/1 on the divert path, BW and PPS, HA solution, RTT, etc.

Distributed Denial of Service

» Traffic diversion (and inspection / cleaning)



Flows

🟥→ “Attack” traffic

🟩→ “Good” traffic

┅→ “Bad” traffic

Router “types”

🟦 Edge

🟧 Access

🟥 Core

IPv6 / MPLS

» **IPv6**

> IPv6 is not the 128 bits address field version of IPv4

> New/updated protocols and new implementations

> Same old and well known bugs will make it into new code

> Current IPv6 “network” is a large lab!

» **Inter-AS MPLS VPNs**

> Multi-Protocol Label Switching is considered as secure as other layer 2 technologies like FR and ATM: but the environment is IP based and much more complex and open

> Inter-Service Provider MPLS VPNs imply transitive trust, no AS boundary anymore

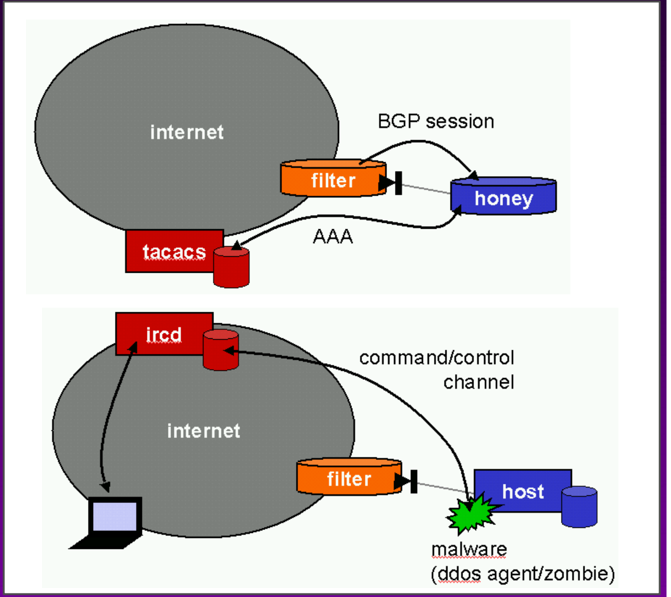
Early Warning System

» **Infrastructure security and EWS**

> DDoS/worms/threats/"IOS upgrade" response process

> "Early Warning System"

- Netflow
- BGP messages accounting
- ACLs logs
- SNMP traps
- Honeyrouters
- Honeybots
- Low interaction honeypots
- Why Honeyspam servers and fighting/patch back with honeypots may be a really bad idea
- Distributed RCs
- nsp-sec, dshield/deepsight, etc.



Lawful Intercept

- **Lawful Intercept**
  - Actively being deployed in lots of countries (ETSI)
  - A cool remote sniffer for Network Operations to dump traffic without having to pray or say “oops!” each time they press “Return” after entering “debug ip packet details” ?
  - An easy way for an attacker to do the same ?
  - The router is not the only device you may have to own, the MD (Mediation Device) is also part of the game

- **Router Traffic Export**

IOS security bugs

» **What if this is only the top of the iceberg…**

> ... and somebody comes up with a bug in the code on the forwarding path ?
>
> - H.323

burning bridges where we can

Phenoelit

> ... and the Cisco IPv4 wedge bug had leaked or been publicly announced ?

> ... and the guys who wrote recent worms had a clue (or different objectives) ?

> “Quick” upgrading Core/Edge vs. bugscrub ?

> Effects/risks of non-diversity (HW and SW) ?

ISPreventer

- » **Engineering / design “issues” and other goodies**
  - > Netgear SNTP “DDos” on WU
  - > Zonelabs’ DNS servers and TAT-14
  - > Verisign’s CRL (and SiteFinder)
  - > b.root-servers.net
  - > RFC1918-like DNS requests and sources (AS112)
- » **(Temporary) filtering**
  - > Do you want to protect the users from the Internet or protect the Internet from end-users ?
  - > NSP/ISP/TierX/BB(Cable, DSL, wLAN) ?
  - > Default permit or default deny ?
  - > How to distribute the filter updates ?
  - > {SCO, MSFT’s WU}.com: DNS “tricks”, filters, etc.

Conclusion

» **Conclusion**

» **See also**

- > Backbone and Infrastructure Security Presentations
  - http://www.securite.org/presentations/secip/
- > (Distributed) Denial of Service Presentations
  - http://www.securite.org/presentations/ddos/

» **Q&A**

Image: http://www.inforamp.net/~dredge/funkycomputercrowd.html