


```
--
^
| /  \ |  \
| \  / |  /
+--> Net::Frame <=> http://www.GomoR.org/ <--+
                        Systems & Security Engineer
                        ---[ zsh$ alias psed='perl -pe ' ]---
                        +--> http://search.cpan.org/~gomor/ <--+
```



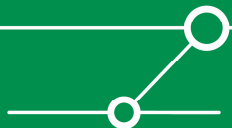
Who we are

■ Dror-John Roecher

- Security Consultant with a faible for enterprise networks and electronic gadgets.
- Based in Germany. Working for ERNW GmbH.
- Check this: www.ernw.de
- no cool nick

■ Patrice <GomoR> Auffret

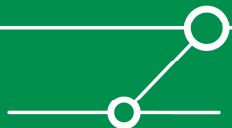
- Security Engineer, Perl network developer
- Author of SinFP (an active and passive OS fingerprinting tool)
- Currently employed by a big service company based in France
- Check this: www.GomoR.org
- And also this:
www.GomoR.org/sinfp



```
--  
^  
| /_/_ |_/_/          http://www.GomOR.org/          <-+  
| \_/_ |_/_/          Systems & Security Engineer          |  
| \_/_ |_/_/          ---[ zsh$ alias psed='perl -pe ' ]---  |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <---+</pre>
```

What we will be talking about...

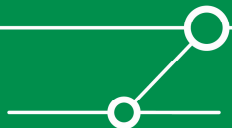
- **Part1 - The (maybe not so) dull theory**
 - The „marketing blah“ – why the stuff we are talking about is important. (very brief!)
 - OSPF operations in some detail.
 - Some ways of breaking OSPF.
 - Mitigating OSPF (again brief)
- **Part2 - The BYOL audience-participation**
 - Show you our tools 😊
 - Attacking OSPF networks




```
--  
^  
| /  \ | /  \      http://www.GomOR.org/      <-+  
| \  / | \  /      Systems & Security Engineer  |  
| \  / | \  /      ---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <---+>
```

Brief History of „Routing Protocol Security“

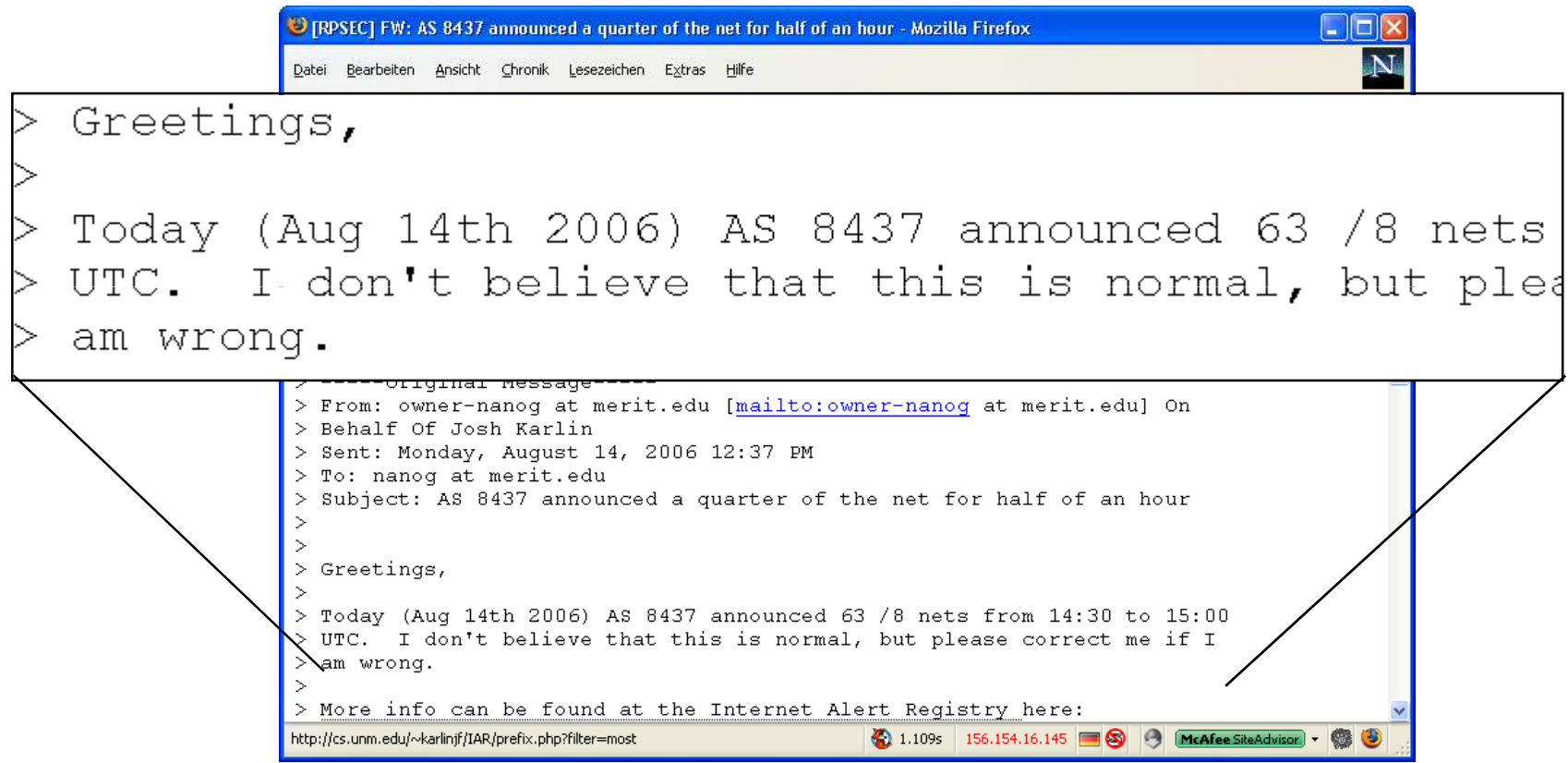
- **Earliest known public discussion: RFC 789, Jan 1981.**
 - Faulty hardware caused faulty network control protocols which in „DoSed“ the ARPANet for a couple of hours...
- **A lot of discussion (with focus on BGP) ever since (just do a google search on „BGP Security“ and be overwhelmed)**
- **Many „add-ons“ [S-BGP, Secure BGP, etc] to BGP – but not much on other protocols.**
- **Structured effort in IETF „rpsec“ working group, but drafts are expired. They are really worth while reading – some guys put a lot of brain into these. Actually the best I have found on the topic so far!**



```
--  
^  
| / _ | /  
| \ / | \  
+--> Net::Frame <=> http://www.GomOR.org/ <--+  
Systems & Security Engineer |  
---[ zsh$ alias psed='perl -pe ' ]--- |  
http://search.cpan.org/~gomor/ <---+>
```



Scary... but fortunately only a „human error“

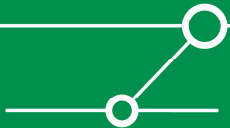



```
--  
^  
| /  |  |  
| \  |  |  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
Systems & Security Engineer |  
---[ zsh$ alias psed='perl -pe ' ]--- |  
http://search.cpan.org/~gomor/ <----+
```



Let's have a look at how OSPF works

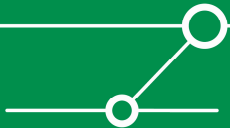
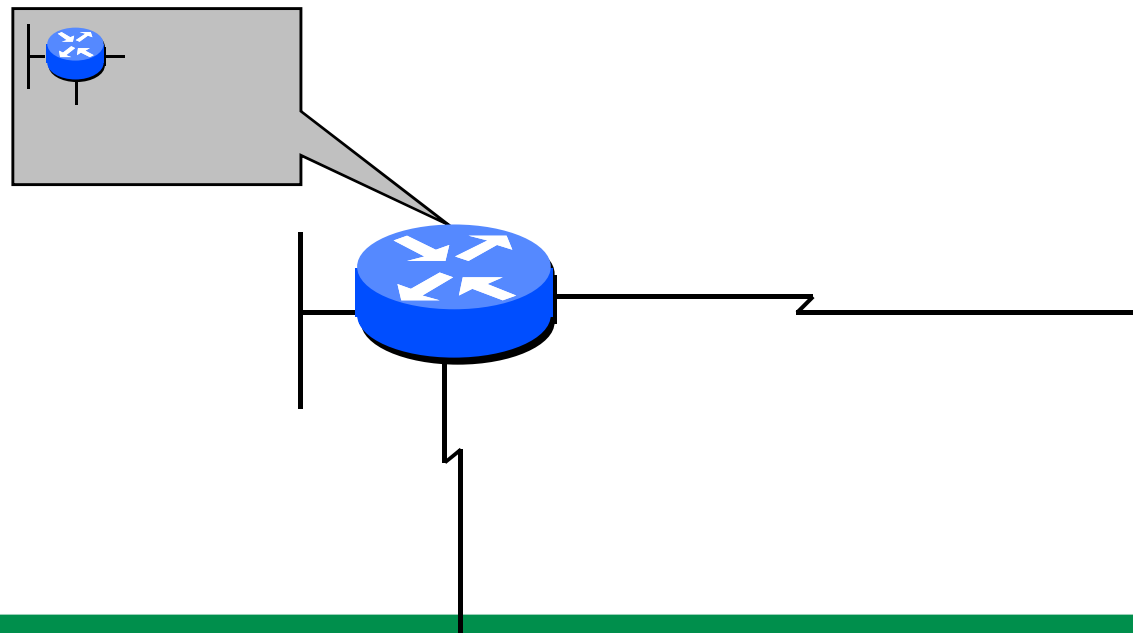
OSPF „quick & dirty“




```
--  
^  
| /_ | /_ |  
| \_ | \_ |  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
Systems & Security Engineer |  
---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> http://search.cpan.org/~gomor/ <---+>
```

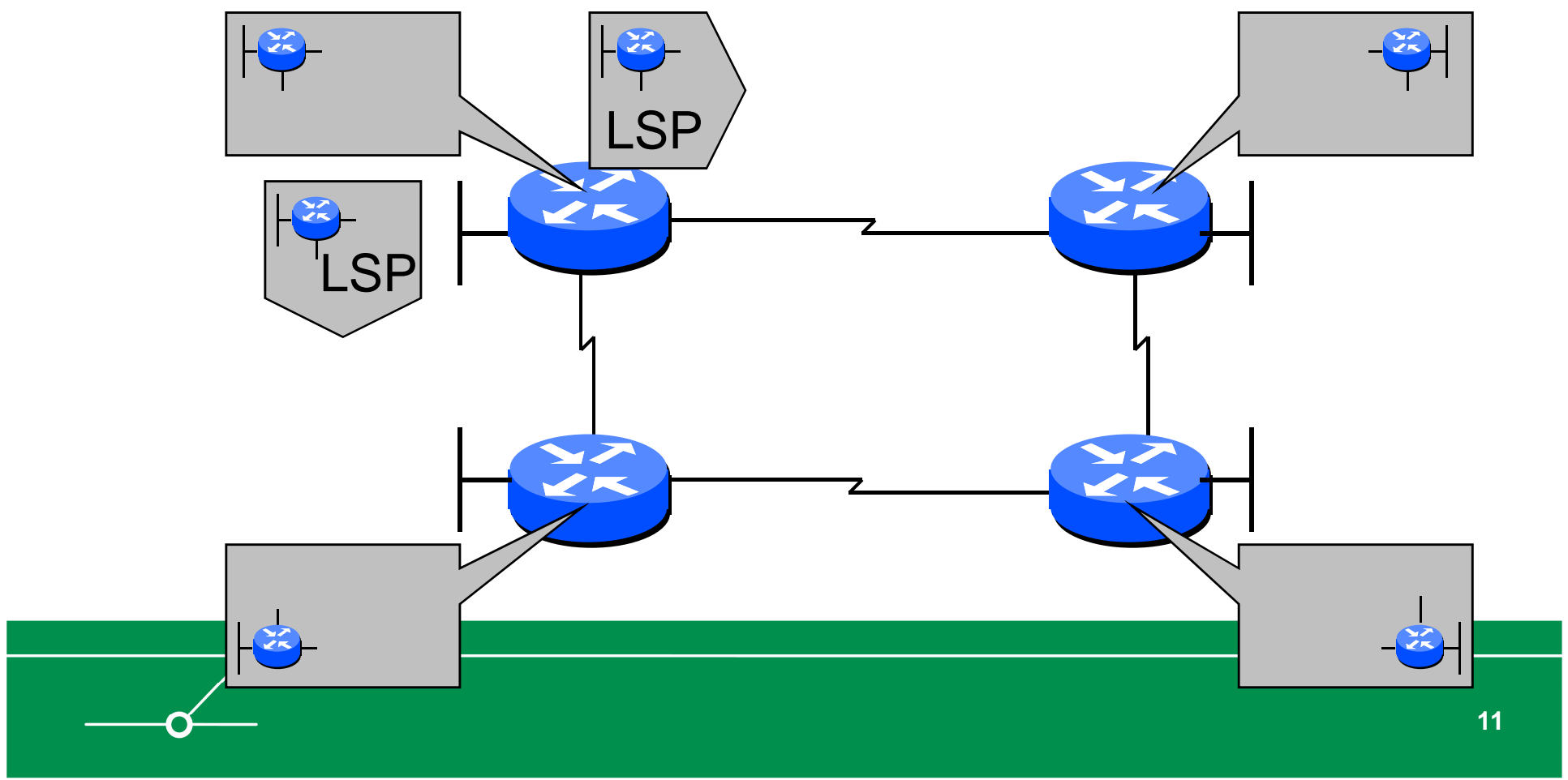
Link State Advertisements

- Every Router advertises its own links.



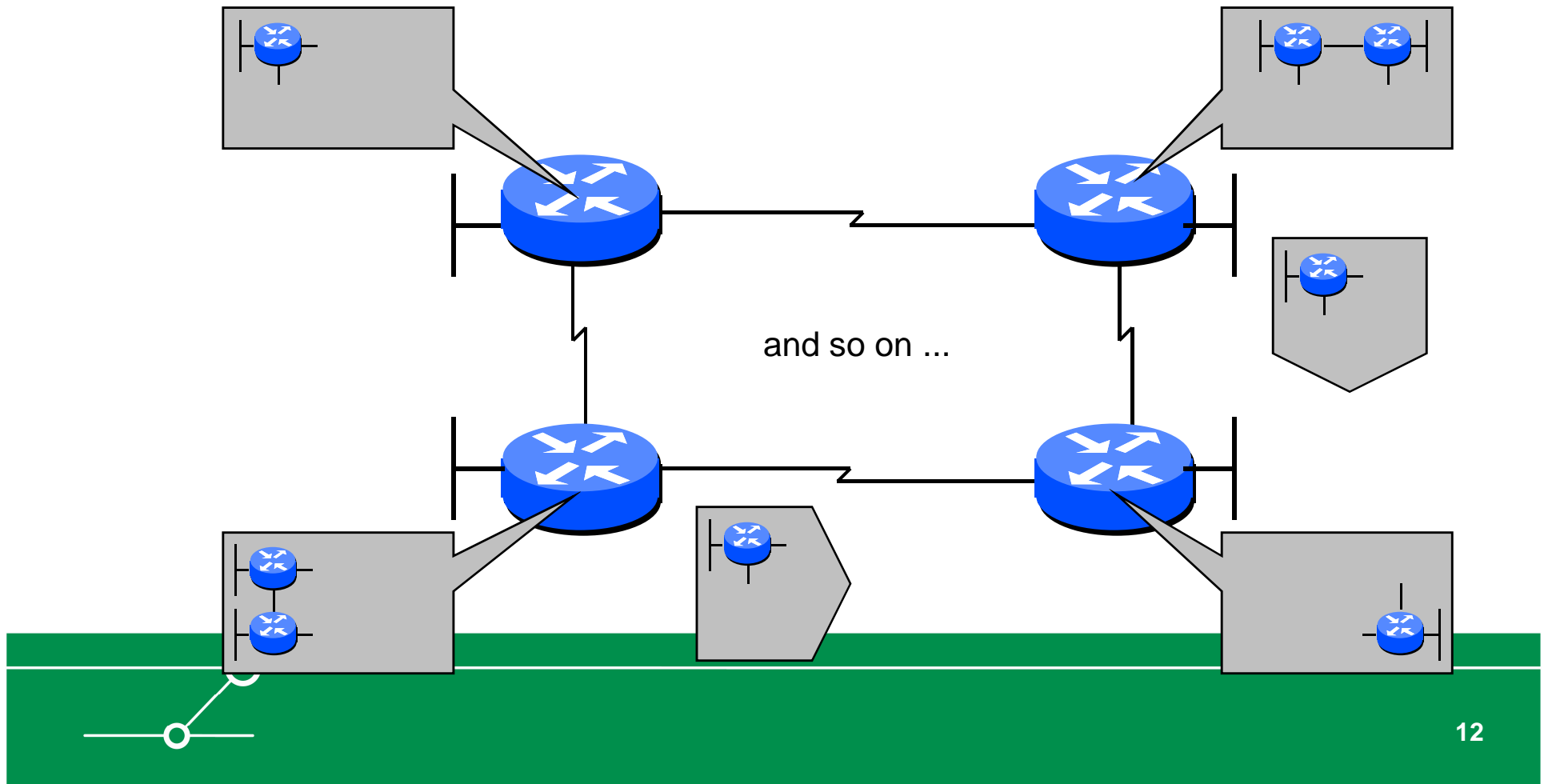
Link State Advertisements

- These LSAs get flooded through the network

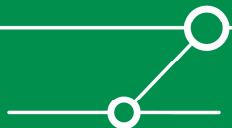
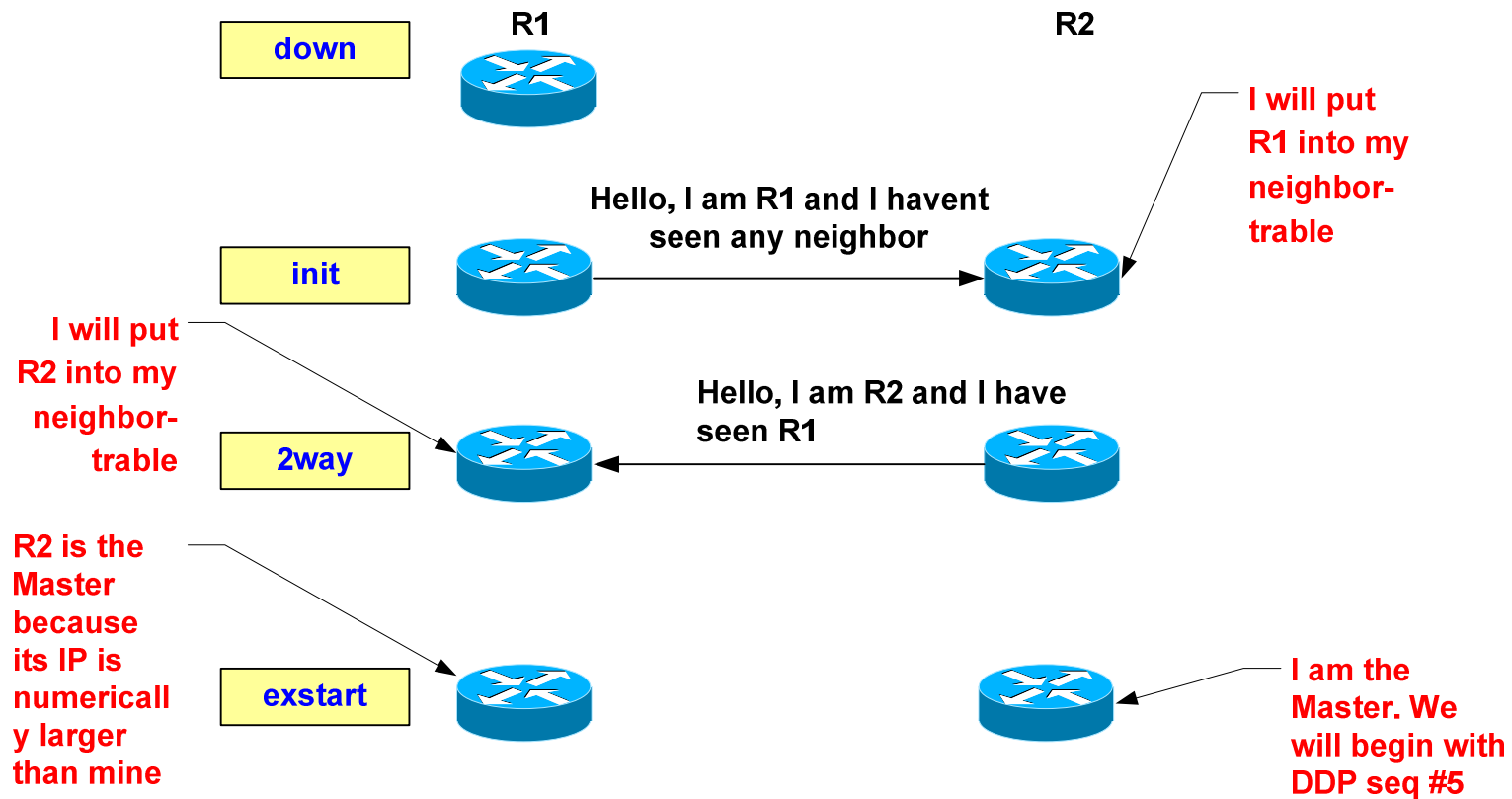


LSA and Flooding

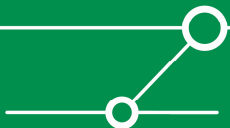
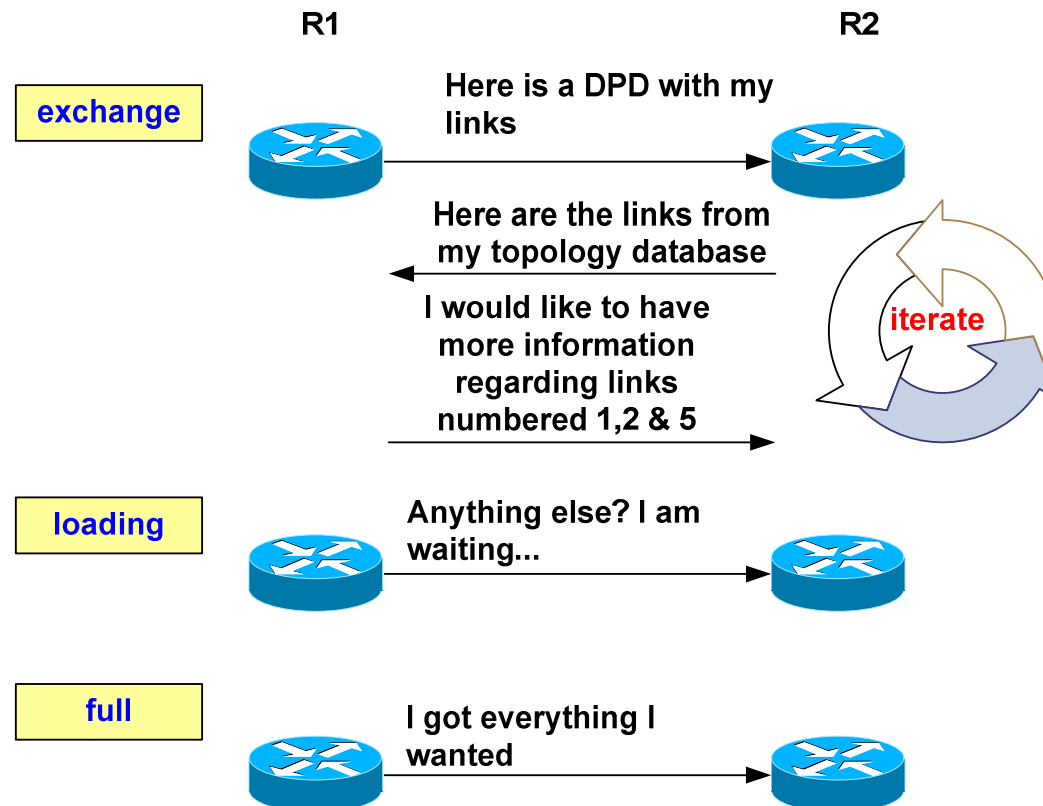
- Every router stores the received LSAs in its topology database



OSPF State Machine (1/2)



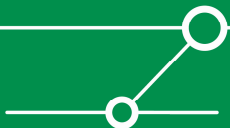
OSPF State Machine (2/2)



```
--  
^  
| /  _ |  _ |  
| \  _ |  _ |  
+--> Net::Frame <=> http://www.GomOR.org/ <--+  
                        Systems & Security Engineer |  
                        ---[ zsh$ alias psed='perl -pe ' ]--- |  
                        +--> http://search.cpan.org/~gomor/ <----+
```

OSPF Authentication

- **Per default OSPF has no authentication.**
- **Two different authentication-schemes exist, which can be used to increase security:**
 - Simple password authentication (that is plaintext passwords)
 - Message Digest authentication (md5 based)
- **Both are based on a „pre shared key“.**



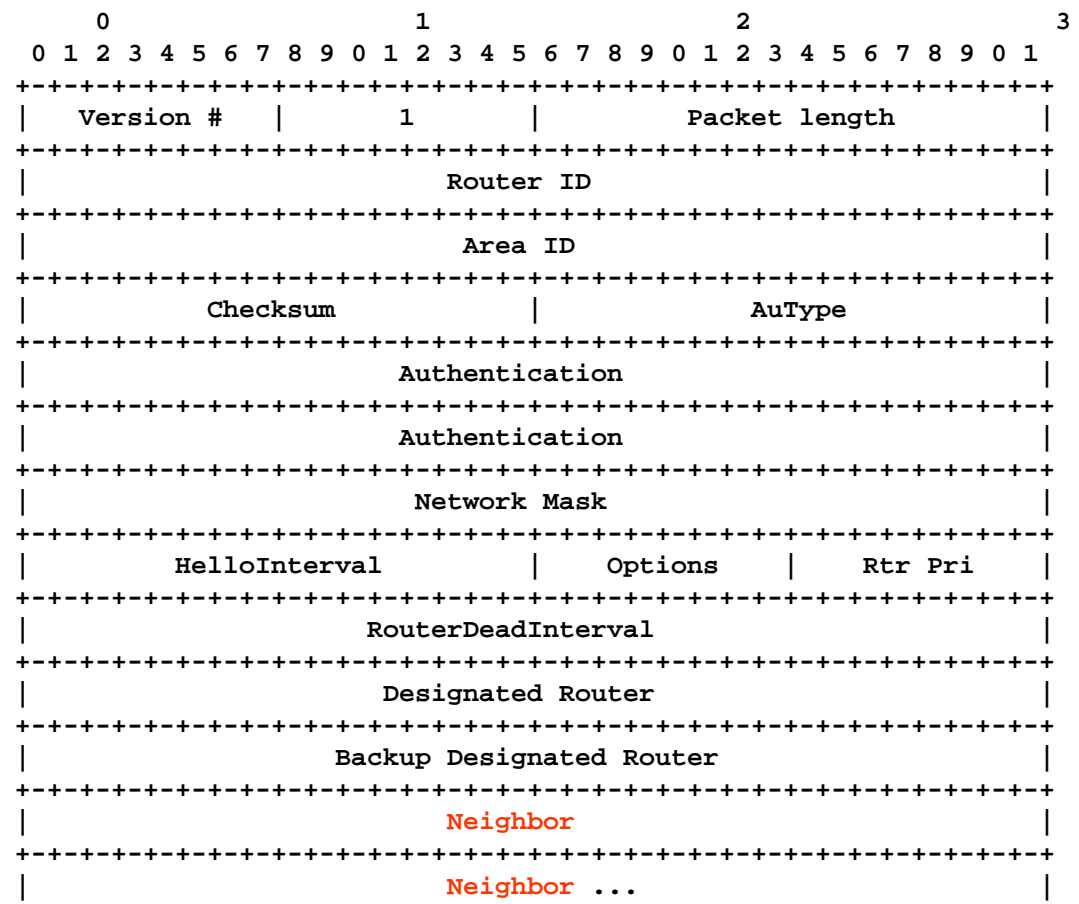
```

--
^
| /  \ |  \
| \  / |  /
+--> Net::Frame <=> http://www.GomR.org/ <--+
                        Systems & Security Engineer |
                        ---[ zsh$ alias psed='perl -pe ' ]--- |
                        +--> http://search.cpan.org/~gomor/ <----+

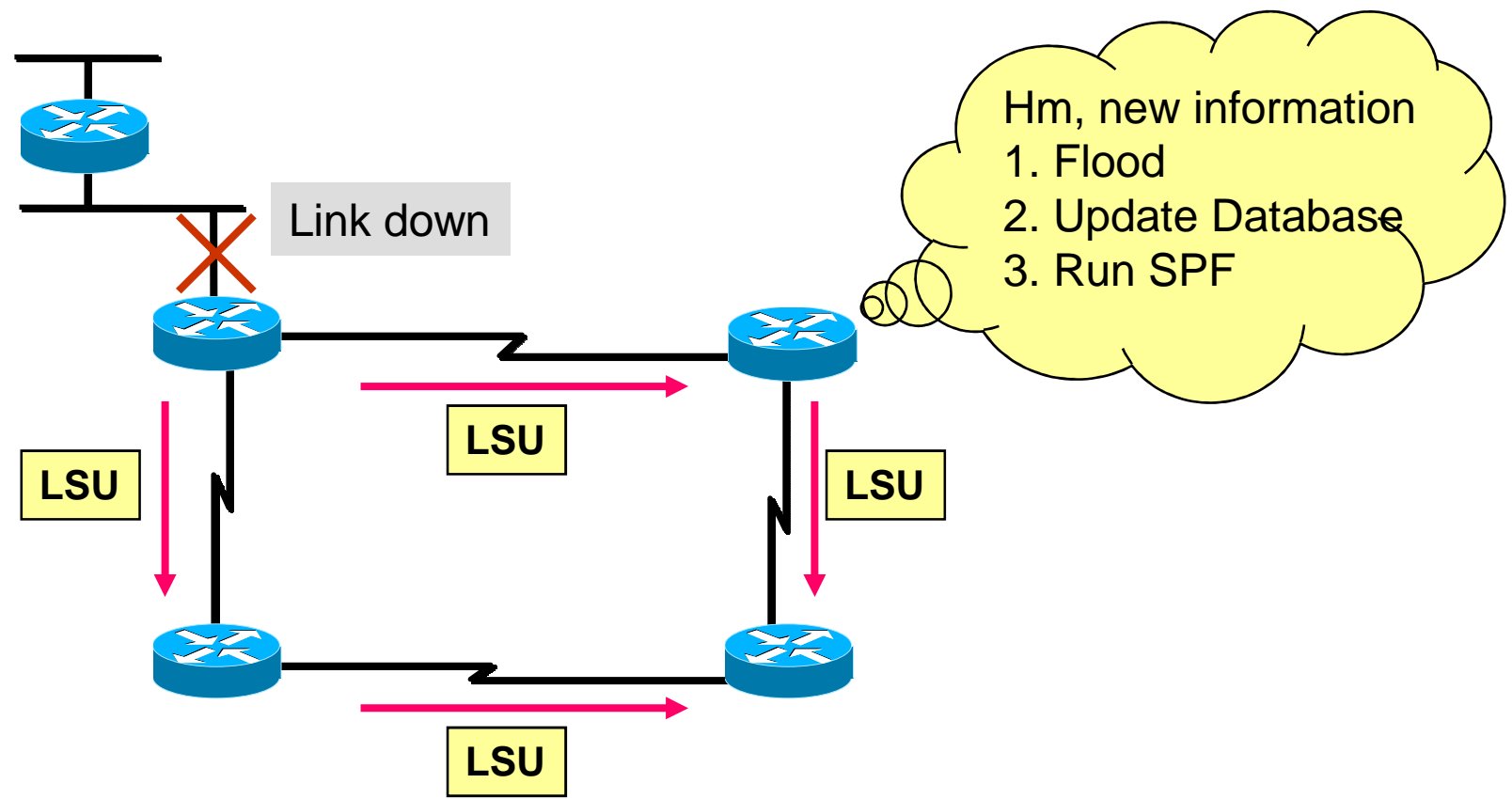
```



Hello Paket Format



Flooding occurs when topology changes are noticed

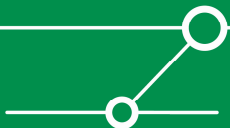



```
--  
^  
| /  \ | /  \      http://www.GomOR.org/      <--+  
| \  / | \  /      Systems & Security Engineer      |  
| /  \ | /  \      ---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <---+>
```



Different Area Types – different information within

- **Normal Area**
 - All LSA Types are forwarded. (The Backbone Area always falls into this category)
- **Stubby Area**
 - No external LSAs are forwarded in stubby areas. Instead a default pointing to the ABR is inserted. Inter area routes are allowed.
- **Totally Stubby Area**
 - No external and no inter area routes – everything that is not local to the area is handled by a default-route.
- **Not So Stubby Area**
 - These area are basically stubby areas with external routes originating from a router within the area.



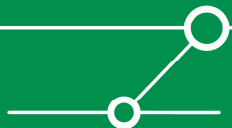
```

--
^
| /  \ |  \ |
| \  / |  / |
+--> Net::Frame <=> http://www.Gomor.org/ <--+
                        Systems & Security Engineer |
                        ---[ zsh$ alias psed='perl -pe ' ]--- |
                        +--> http://search.cpan.org/~gomor/ <----+

```

Different LSA for different information....

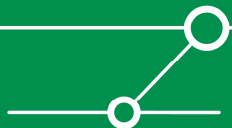
LSA-Type	who?	Content?
Type1:	everyone	Links
Type2:	DR	Network
Type3:	ABR	Network Summaries (interarea)
Type4:	ABR	Routes to the ASBR
Type5:	ASBR	External Routes
Type7:	ASBR	NSSA External Routes (Type7-LSAs are converted by ABRs to Type5-LSAs).

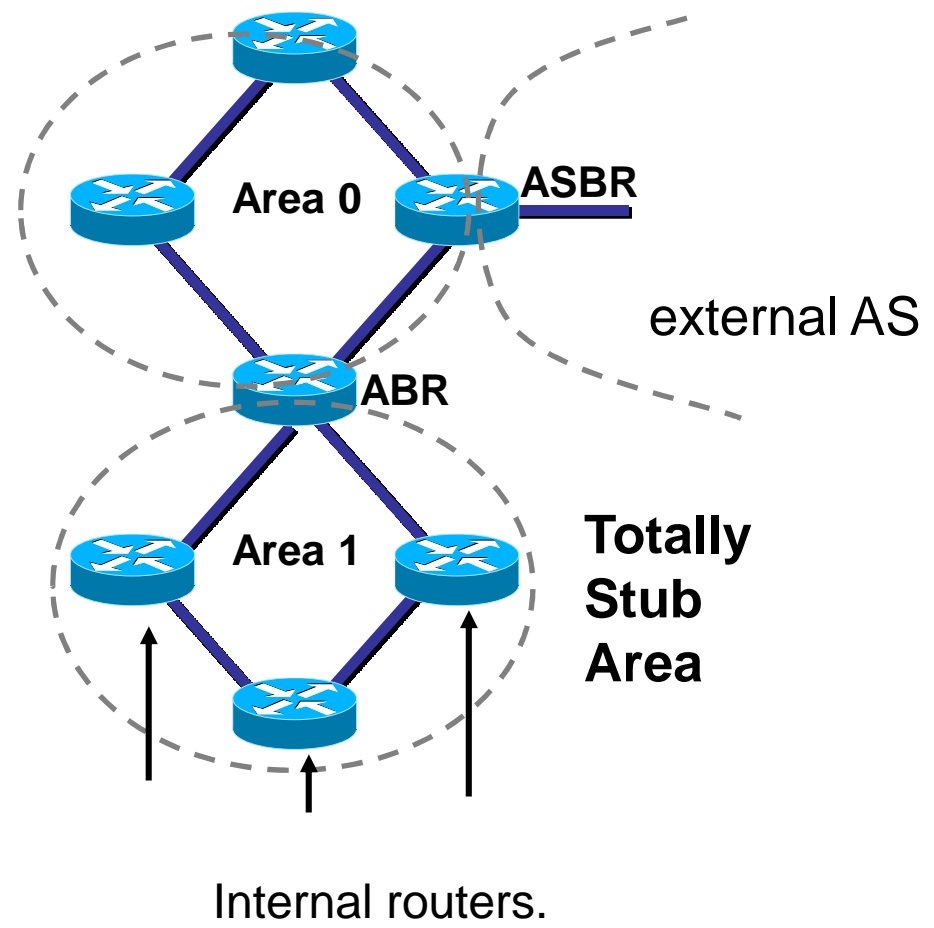



```
--
^
| /  \ | /  \
| \  / | \  /
+--> Net::Frame <=> http://www.GomOR.org/ <--+
                        Systems & Security Engineer
                        ---[ zsh$ alias psed='perl -pe ' ]---
                        +--> http://search.cpan.org/~gomor/ <--+
```

LSA Types

- **ASBR Summary (Type4-LSA)**
- **LSA Type4** are generated by **ABRs** and include routes to the ASBRs).
- **ASBR External LSA (Type5-LSA,Type7-LSA)**
- **ASBRs** send ASBR External LSAs (**Type5-LSA**), including information about networks outside the OSPF AS or a default route to outside the OSPF AS.
- If these Type-5 LSAs are sourced by an ASBR of a NSS, it is send as a **Type7-LSA**. Type7-LSAs are changed to **Type5-LSAs** by the ABR of the NSSA.



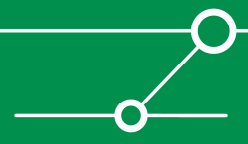


Totally stub Area

Totally stubby area

No external routes and not inter-area routes are known within a totally stubby area.

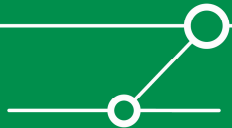
Everything which is not local to the area is routed via a default to an ABR.




```
--  
^  
| /  | /  
| \  | \  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
Systems & Security Engineer |  
---[ zsh$ alias psed='perl -pe ' ]--- |  
http://search.cpan.org/~gomor/ <----+
```



Attacking OSPF

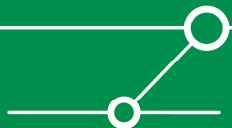


```
--  
^  
| /  _ |  _ |  
| \  _ |  _ |  
+--> Net::Frame <=> http://www.GomOR.org/ <--+  
                        Systems & Security Engineer |  
                        ---[ zsh$ alias psed='perl -pe ' ]--- |  
                        http://search.cpan.org/~gomor/ <----+
```



What are the consequences of attacking OSPF?

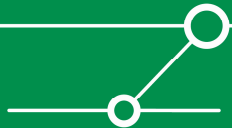
- **Disruption and/or Manipulation of the Routing Domain**



```
--  
^  
| /  |  |  
| \  |  |  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
                        Systems & Security Engineer |  
                        ---[ zsh$ alias psed='perl -pe ' ]--- |  
                        http://search.cpan.org/~gomor/ <-----+
```



Attack Vectors



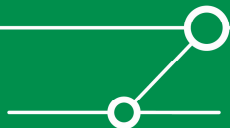
```
--
^
| /  \ |  \
| \  / |  /
+--> Net::Frame <=> http://www.GomoR.org/ <--+
                        Systems & Security Engineer
                        ---[ zsh$ alias psed='perl -pe ' ]---
                        +--> http://search.cpan.org/~gomor/ <----+
```

OSPF Attack Vectors...

■ Classification of attack-vectors:

- Attacks which originate from the outside of the OSPF network
 - Prerequisite: Attacker is able to send unicast OSPF-packets to an internal OSPF router. This should not be possible, because OSPF packets should not be allowed to enter the network.
- Attacks which originate from the inside of the OSPF network
 - **Device Compromise:** Attacker has administrative access (console or ssh) to an OSPF-router.
 - **Link Compromise:** Attacker has access to a network-link, where OSPF is being spoken by one or more connected routers.
- Attacks through „broken“ implementations: BOs in ospfd etc. – not in scope for today's talk, even though they may have a huge impact on overall security.

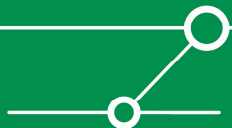
That is what we will talk about today :-)



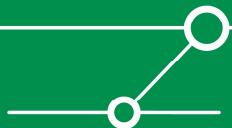
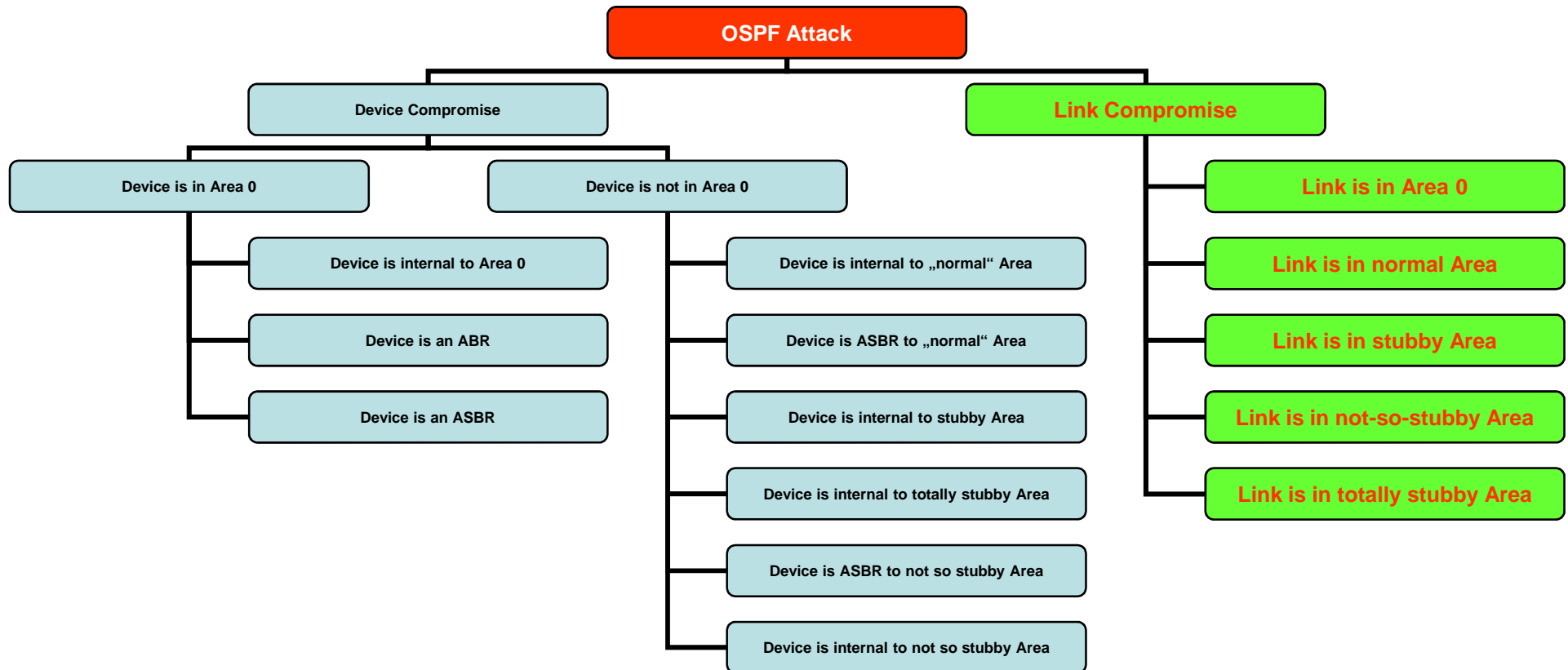
```
--  
^  
| /  | /  
| \  | \  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
                        Systems & Security Engineer |  
                        ---[ zsh$ alias psed='perl -pe ' ]--- |  
                        http://search.cpan.org/~gomor/ <----+
```

Link Compromise

- **Link is in Area 0**
- **Link is not in Area 0**
 - Link is in „normal“ Area
 - Link is in „stubby“ Area
 - Link is in „not so stubby“ Area
 - Link it in „totally stubby“ Area



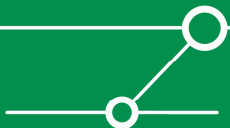
The Attack Vectors as a graph



```
--  
^  
| /  _ |  _ |  
| \  _ |  _ |  
+--> Net::Frame <=> http://www.GomOR.org/ <--+  
                        Systems & Security Engineer |  
                        ---[ zsh$ alias psed='perl -pe ' ]--- |  
                        http://search.cpan.org/~gomor/ <----+
```

Some Threats through Device Compromise

- **We will not go into depth here (mostly for time-reasons and because threats are somewhat obvious).**
- **Some possible threats:**
 - DoS: Dropping of routes
 - DoS: (Partial) Disabling of OSPF
 - DoS: Addition of „bogus“ routes via loopback interfaces (e.g. with /32 mask to have a „longest match“)
 - DoS: Creating Routing loops (which adds congestion besides DoS)
- **These are not very interesting, because any change to OSPF will affect the local routing table, too and the interesting attacks avoid just that.**



```
--  
^  
| /  \ | /  \      http://www.GomoR.org/      <--+  
| \  / | \  /      Systems & Security Engineer      |  
| \  / | \  /      ---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <---+>
```



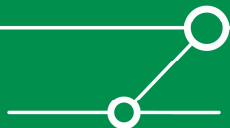
Threats through Link Compromise

■ Denial of Service:

- Blackhole: Traffic is directed to a router which cannot handle the load.
- Starvation: Traffic is forwarded to a part of the network, that can not deliver it.
- Delay: Traffic is routed via a suboptimal path.
- Loop: Traffic is forwarded along a looping path.
- Partition: Some part of the network believes it is partitioned from the rest, when in fact it is not.
- Churn: Forwarding on the network changes rapidly, resulting in large variations of data-delivery patterns (impacting congestion control mechanisms).
- Instability: OSPF itself becomes unstable so that global convergence is never achieved.
- Overload: OSPF messages themselves become a significant part of the network traffic.
- Resource Exhaustion: OSPF messages cause exhaustion of router resources (queues, memory, cpu).

■ Man in the Middle

- Eavesdropping: Carefully crafted insertion of routing information may lead to rerouting through attacker which may put the attacker in the packet-path. These are quite difficult to accomplish. But this is (imho) the most interesting attack scenario.

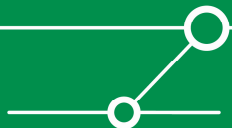


```
--  
^  
| /  \ | /  \      http://www.GomOR.org/      <-+  
| \  / | \  /      Systems & Security Engineer      |  
| /  \ | /  \      ---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <----+
```

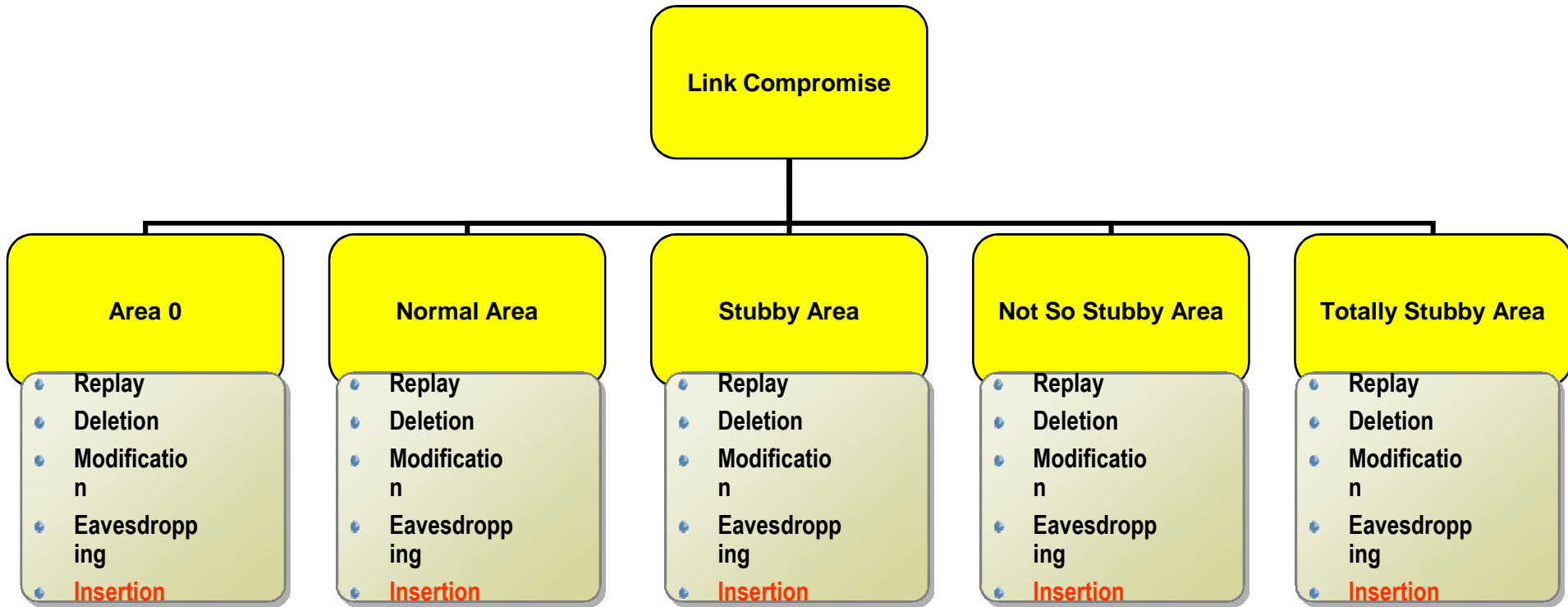


Attacks on „Link Compromise“ fall into one of these classes

- **Message Replay**
- **Message Insertion (that will be the focus today)**
- **Message Deletion (usually detectable by the sender)**
- **Message Modification**
- **Message Eavesdropping (almost always needed to gain some knowledge about how OSPF is set up)**



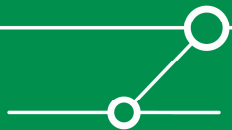
Link Compromise




```
--  
^  
| /  \ | /  \      http://www.GomOR.org/      <-+  
| \  / | \  /      Systems & Security Engineer      |  
| \  / | \  /      ---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <----+
```



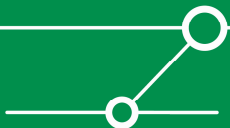
Attack Classification - Message Insertion



```
--  
^  
| /  \ | /  \  
| \  / | \  /  
+--> Net::Frame <=> http://www.GomOR.org/ <--+  
                        Systems & Security Engineer |  
                        ---[ zsh$ alias psed='perl -pe ' ]--- |  
                        http://search.cpan.org/~gomor/ <---+>
```

Categories of Attacks – Message Insertion (1/2)

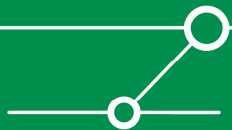
- **Setting up phanthom routers (routers that dont exist)**
 - Simple „hello“ suffices to get into neighbor-tables. But that should have no impact – just a „gimmick“
- **Spoofing messages from existing routers**
 - Send „hellos“ with on a link where the router acutally isnt located (not sure if OSPF fightback should come into place).
 - Send „hellos“ on a link where the router is located
 - Send spoofed LSAs (here the OSPF fightback mechanism should come into place) – which can be leveraged for DoS by taking advantage of timer-mechnisms in OSPF.




```
--  
^  
| /  | /  
| \  | \  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
Systems & Security Engineer |  
---[ zsh$ alias psed='perl -pe ' ]--- |  
http://search.cpan.org/~gomor/ <----+
```



Mitigation




```
--  
^  
| /  \ | /  \      http://www.GomOR.org/      <-+  
| \  / | \  /      Systems & Security Engineer      |  
| \  / | \  /      ---[ zsh$ alias psed='perl -pe ' ]--- |  
+--> Net::Frame <=> http://search.cpan.org/~gomor/ <----+
```

Prerequisites for BYOL

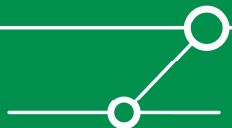
■ Technical

- A networked Laptop with VMWare Workstation or Server installed
- Our prepared VMWare-Image

■ Knowledge & Experience

- Some knowledge of Linux & Perl
- Some experience with Cisco IOS

- And **please follow the instructions**, the lab is quite complex and we want to avoid total chaos.



```
--  
^  
| /  | /  
| \  | \  
+--> Net::Frame <=> http://www.GomOR.org/ <-+  
Systems & Security Engineer |  
---[ zsh$ alias psed='perl -pe ' ]--- |  
http://search.cpan.org/~gomor/ <-----+
```



děkuji pěkně

dotazy a že odpovědi...

