Flow-Level Traffic Analysis of the Blaster and Sobig Worm Outbreaks in an Internet Backbone

Thomas Dübendorfer, Arno Wagner,
Theus Hossmann, Bernhard Plattner
ETH Zurich, Switzerland

duebendorfer@tik.ee.ethz.ch

DIMVA 2005, Wien, Austria

TIK
Computer Engineering and Networks Laboratory

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Agenda

1) Introduction

2) Flow-Level Backbone Traffic

3) Network Worm Blaster.A

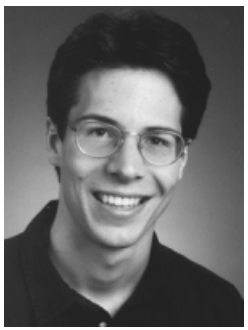
4) E-Mail Worm Sobig.F

5) Conclusions and Outlook

Authors

Prof. Dr. Bernhard Plattner

Professor, ETH Zurich (since 1988)
Head of the Communication Systems Group at the Computer Engineering and Networks Laboratory TIK
Prorector of education at ETH Zurich (since 2005)



Thomas Dübendorfer

Dipl. Informatik-Ing., ETH Zurich, Switzerland (2001)
ISC^2 CISSP (Certified Information System Security Professional) (2003)
PhD student at TIK, ETH Zurich (since 2001)
Network security research in the context of the **DDoSVax** project at ETH

Further authors: **Arno Wagner**, **Theus Hossmann**

Worm Analysis

Why analyse Internet worms?

- basis for research and development of:
  - worm detection methods
  - effective countermeasures
- understand network impact of worms

Wasn‘t this already done by anti-virus software vendors?

- Anti-virus software works with *host-centric* signatures

Research method used

1. Execute worm code in an Internet-like **testbed** and observe infections
2. Measure **packet-level** traffic and determine *network-centric* worm signatures on flow-level
3. Extensive analysis of **flow-level** traffic of the actual worm outbreaks captured in a Swiss backbone

Related Work

Internet backbone worm analyses:

- Many **theoretical** worm spreading **models** and simulations exist (e.g. for Code Red)
- **CAIDA's Network Telescope**: Code Red, Slammer, Witty (observation of e-mail worms and multi-stage worms is impossible with such a **passive blackhole monitoring system**)
- **ETH's DDoSVax project**: Blaster, Sobig.F et al.

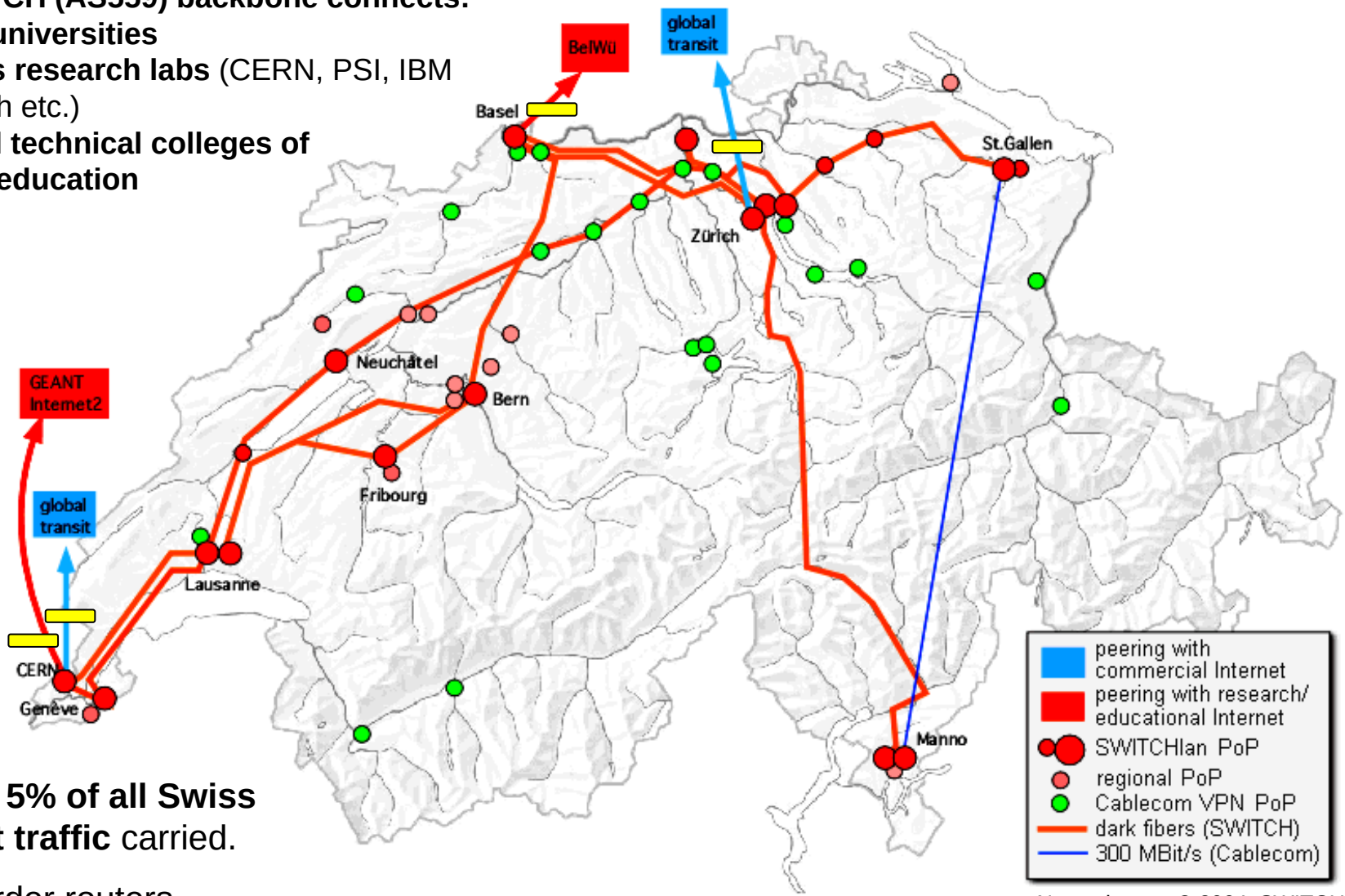
→ Worm analyses based on Internet backbone traffic are *very* rare

AS559 Backbone

SWITCH

The SWITCH (AS559) backbone connects:

- **Swiss universities**
- **Various research labs** (CERN, PSI, IBM research etc.)
- **Federal technical colleges of higher education**



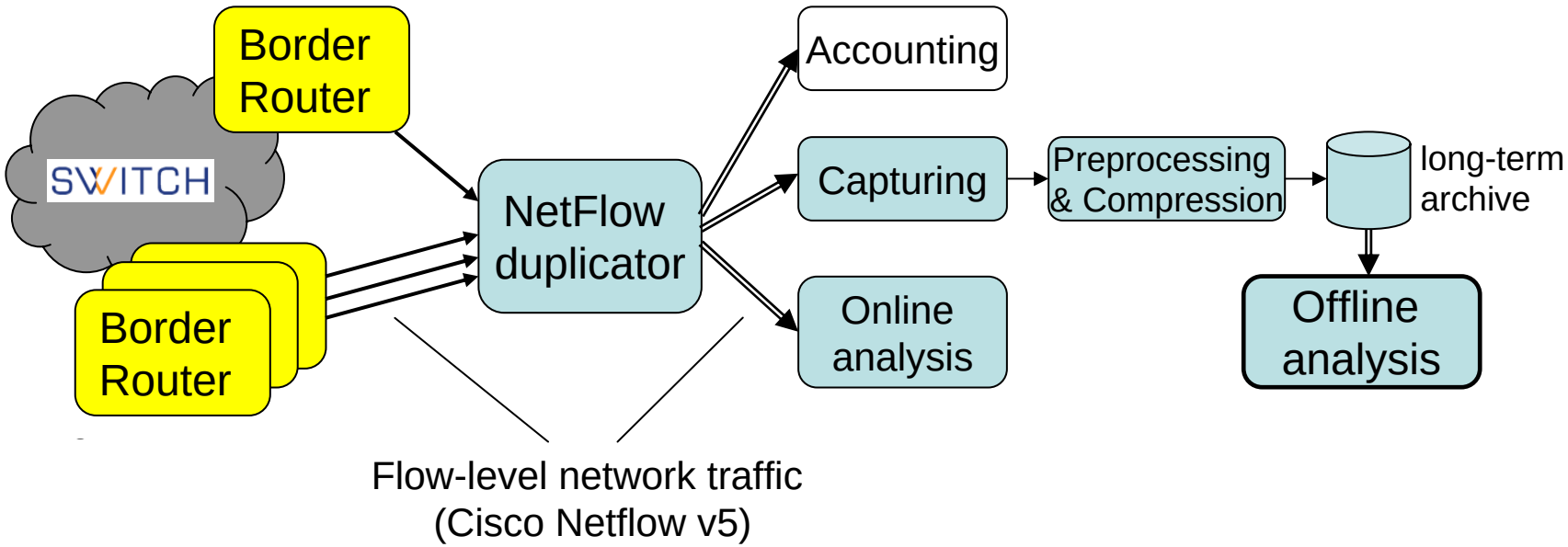
Network map: © 2004, SWITCH

Approx. **5% of all Swiss Internet traffic** carried.

Border routers

DDoSVax Infrastructure

Flow-level traffic acquisition and analysis (simplified):



DDoSVax collaboration and funding partners:

SWITCH FNSNF

DDoSVax … “In Search of a ***Vacc***ine against ***DDoS*** attacks”

„Flow“ Definition

Flow ≈ Stream of sequential related IP packets
Example: 109.132.30.30:80 -> 80.82.130.100:1230 TCP 40 packets 80‘556 bytes

An Internet traffic „**flow**“ is defined as
- a unidirectional stream of IP packets
- between two hosts (i.e. source and destination IP address)
- using the same protocol (TCP, UDP, ICMP, others)
- with a fixed source and destination port (for TCP, UDP)
- using the same routing parameters (router in-/output interfaces)

A flow contains **no payload**, but gives:
- number of bytes
- number of packets
- start and end time of the flow (in milliseconds)
- some other (mostly routing related) information

A flow ends upon timeout conditions or upon stream end (TCP FIN).

We use **CISCO‘s** popular **NetFlow** v5 format (48 bytes per flow record).

DDoSVax Statistics

The **DDoSVax traffic archive** contains the complete unsampled flow-level (NetFlow v5) AS559 border router traffic since early 2003 in bzip2 compressed form:

~17 Gigabytes/day

~6 Terabytes/year

A one hour DDoSVax flow-level trace of the AS559 border routers during a working day contains:

~60 million flows (NetFlow v5)

~200‘000 active AS559-internal hosts

~800‘000 active AS559-external hosts

Agenda

1) Introduction

2) Flow-Level Backbone Traffic

→ 3) Network Worm Blaster.A

4) E-Mail Worm Sobig.F

5) Conclusions and Outlook

Blaster Worm

Blaster.A network worm:

- Outbreak on Monday, August 11th, 2003, 16:35 UTC
- 200‘000 (Internet Storm Center) – 8 mill. (Microsoft) infected computers
- exploits remote procedure call (RPC) DCOM buffer overflow in Microsoft Windows 2000/XP on port 135/TCP known since July 2003
- Impact: Internet resource misuse for spreading; reboot of unpatched Win XP systems; (unsuccessful) DDoS attack on windowsupdate.com; host infections

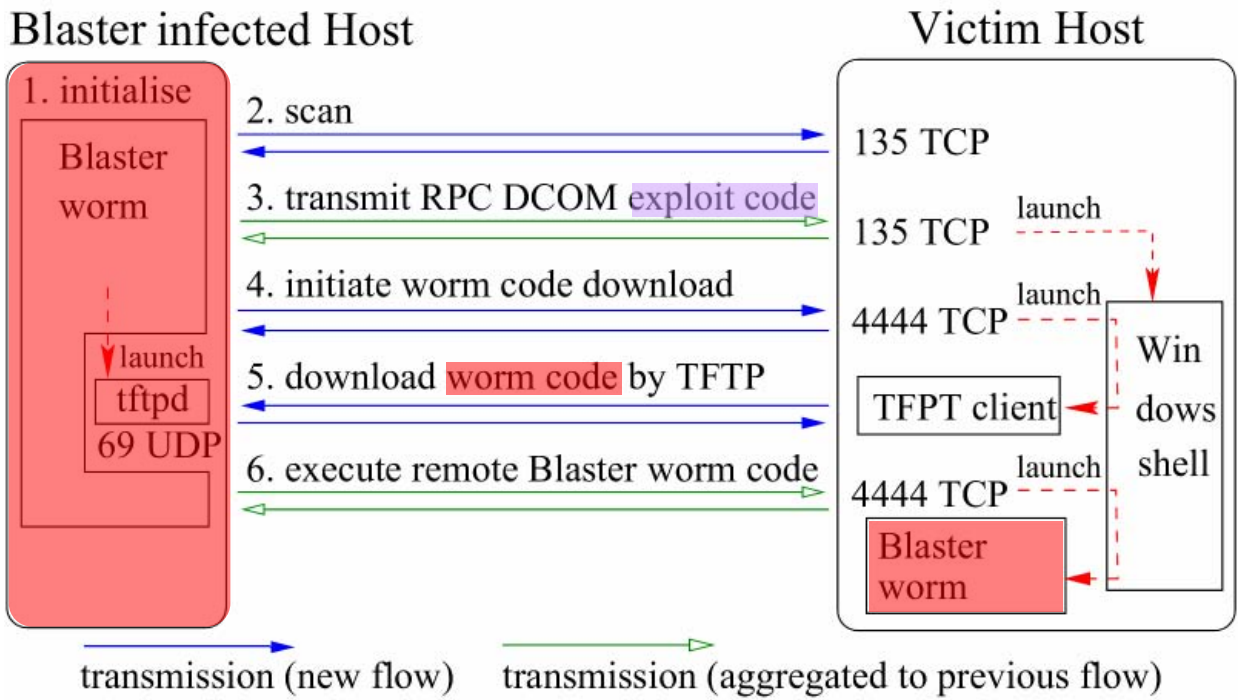
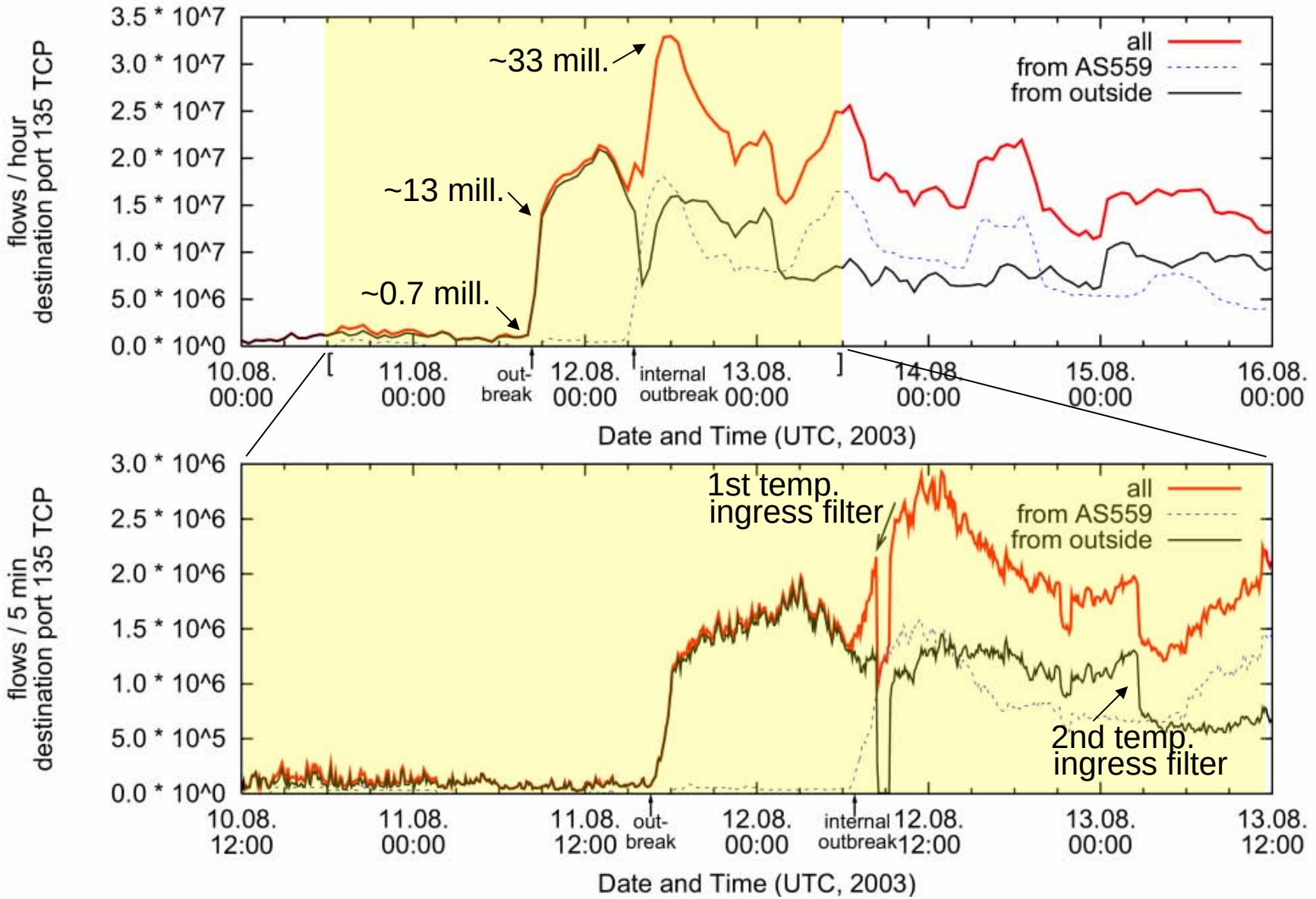
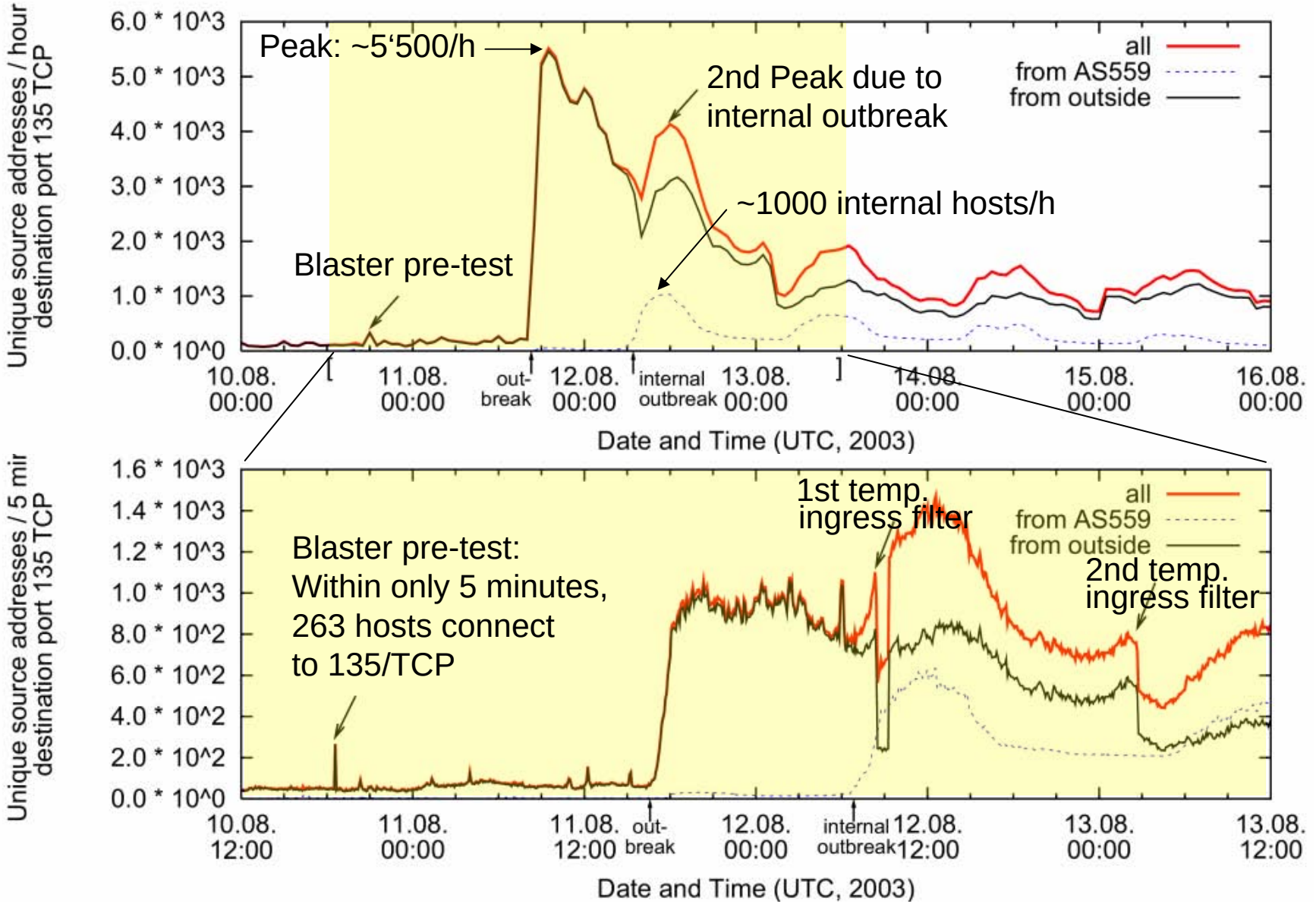


Figure: A network centric view on Blaster‘s infection steps

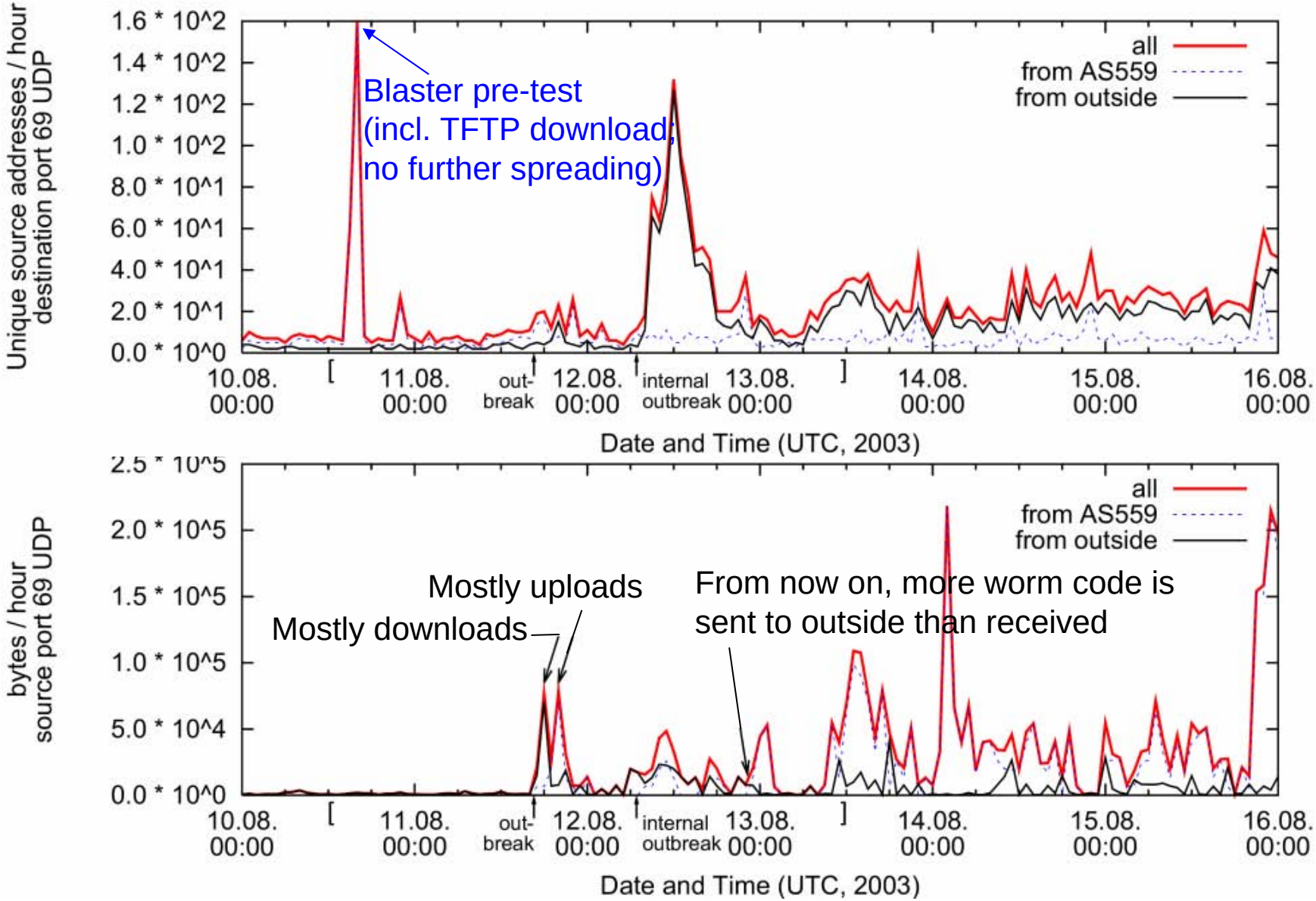
Flows to 135/TCP



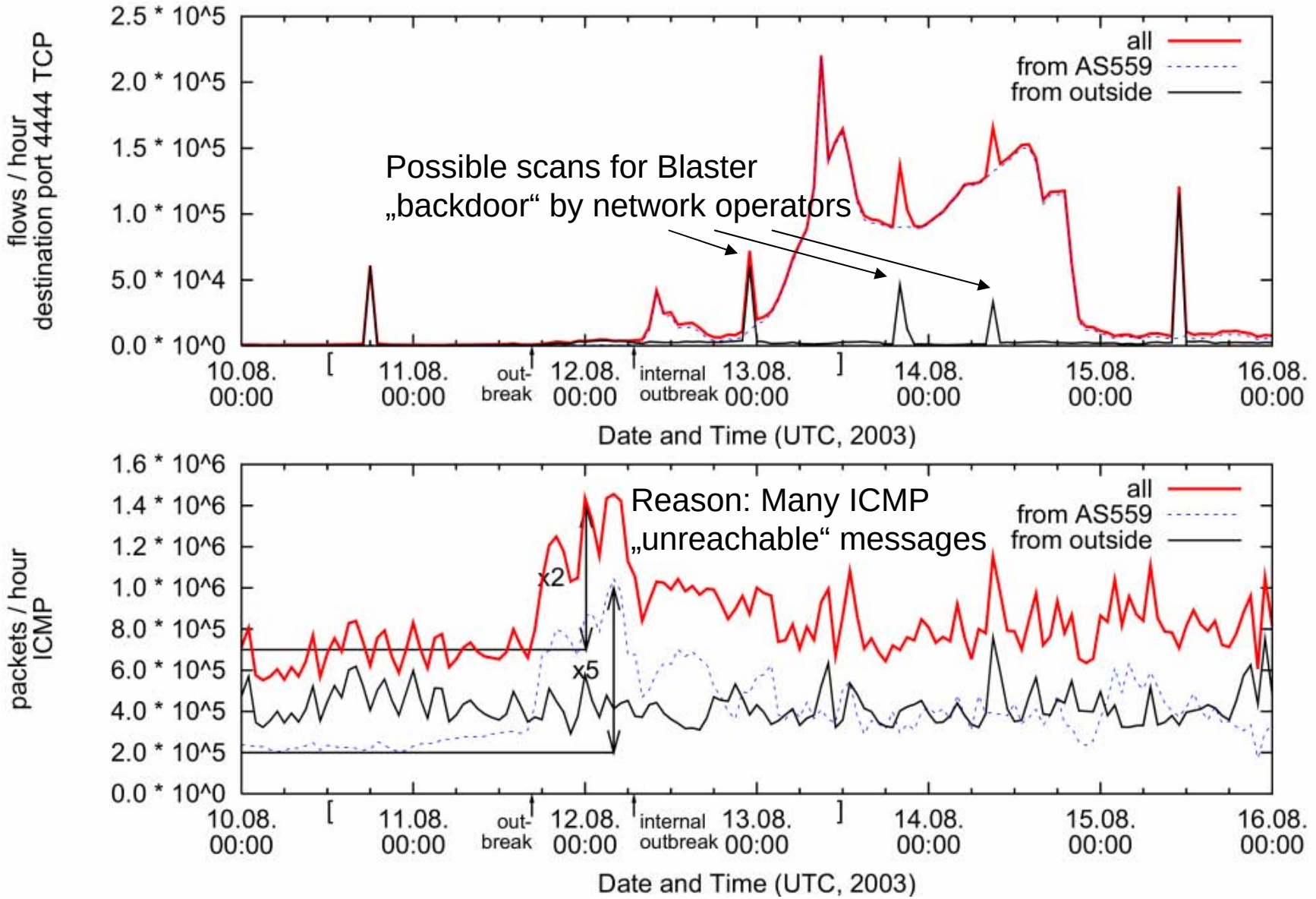
Unique Source Addresses to 135/TCP



69/UDP activity



4444/TCP and ICMP activity



Blaster‘s Infection Attempts

Infection stages:

A) No response from victim upon connection request to 135/TCP.

B) Response from victim, but port 135/TCP is closed.

C) Victim receives exploit code but was not vulnerable or wrong exploit code was sent (80% WinXP, 20% Win 2000).

D) Victim receives and executes exploit code but no worm code is downloaded.

E) Victim is successfully infected.

Stage	135/TCP		4444/TCP		69/UDP	
	A→V	A←V	A→V	A←V	A←V	A→V
A	■	-	-	-	-	-
B	■	■	-	-	-	-
C	■	■	■	■	-	-
D	■	■	■	■	■	-
E	■	■	■	■	■	■

Legend:

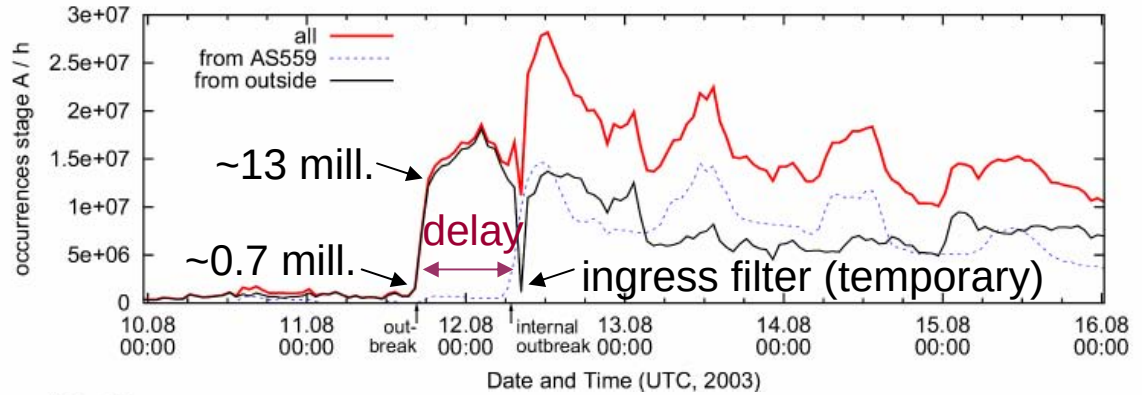
A(ttacker) → V(ictim)

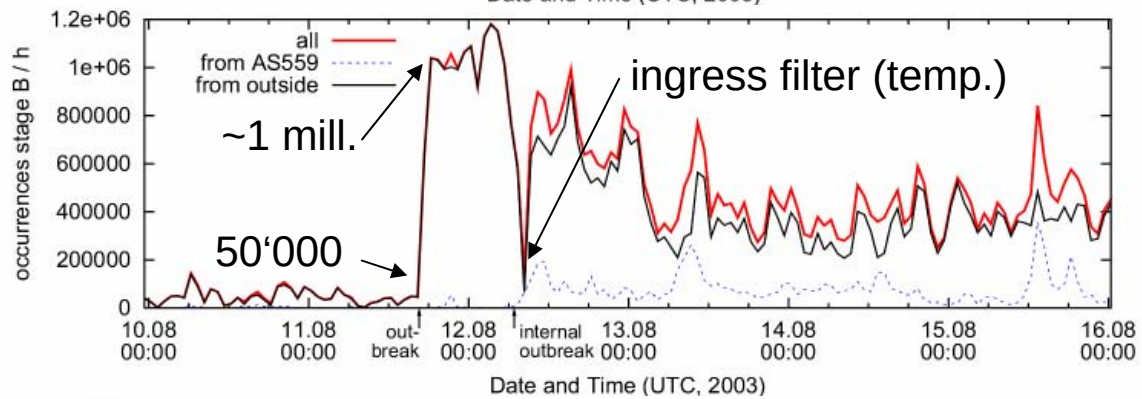
V(ictim) ← A(ttacker)

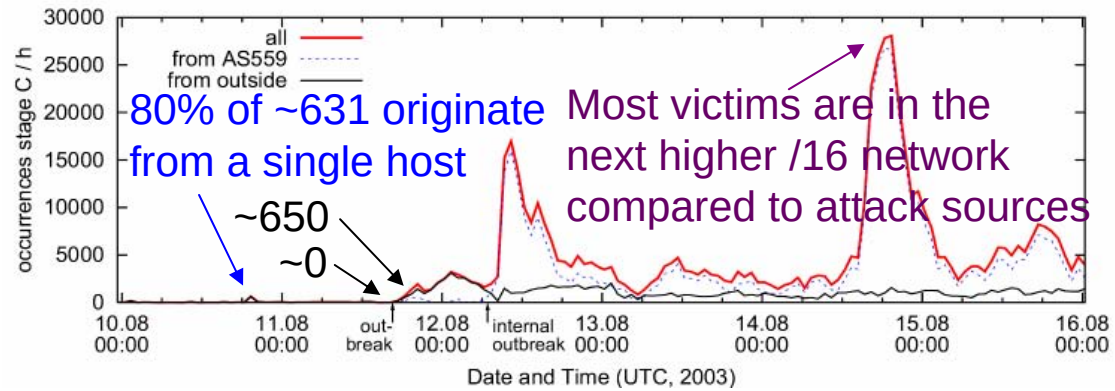
■ flow required

Figure: Flows required for Blaster‘s infection stages A - E

Infection Attempt Stages A, B, C







Infection stages:

A) No response from victim upon connection request to 135/TCP.

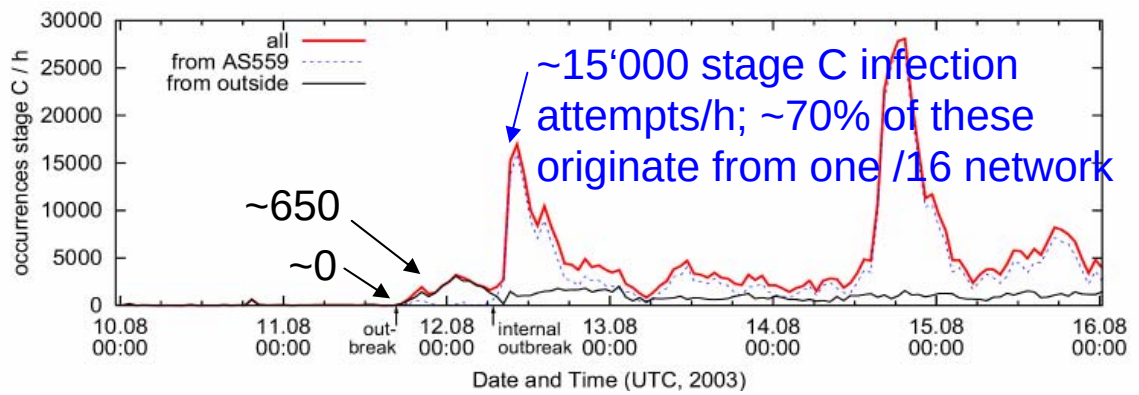
8/11 16:35 UTC external outbreak

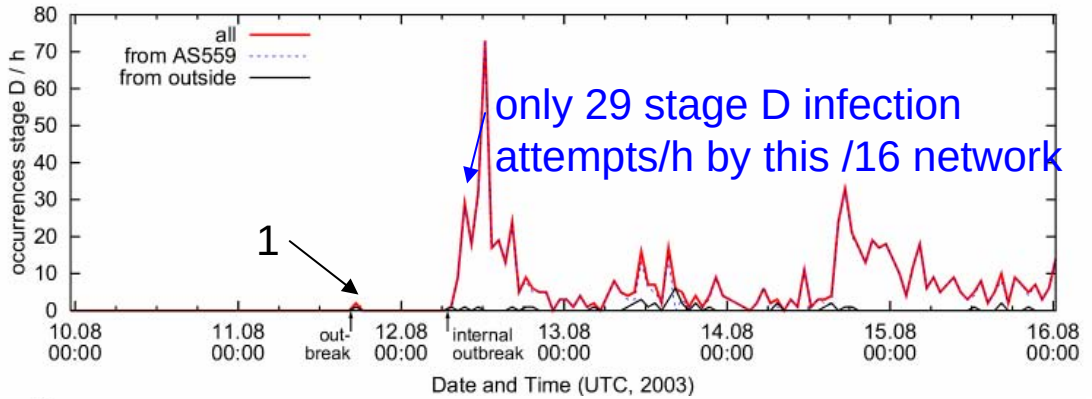
8/12 6:50 UTC internal outbreak

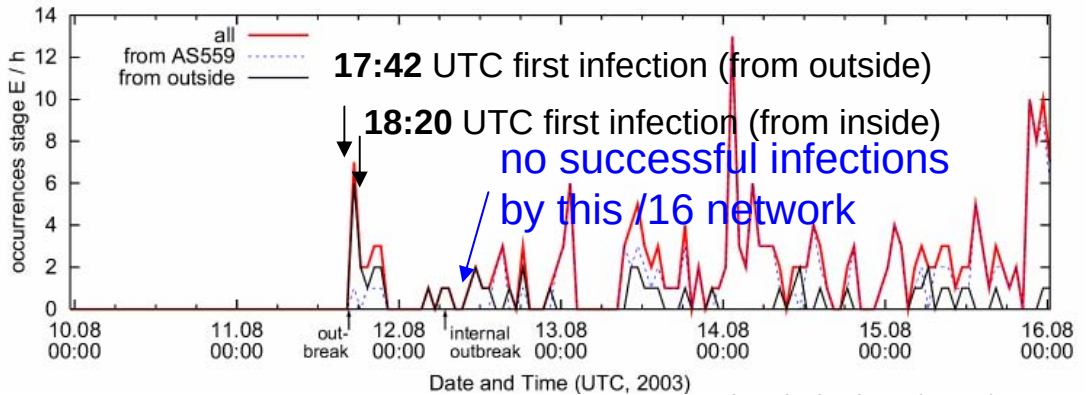
B) Response from victim, but port 135/TCP is closed.

C) Victim receives exploit code but vulnerability was not present or wrong exploit code was sent.

Infection Attempt Stages C, D, E







Infection stages (cont‘d):

C) Victim receives exploit code but was not vulnerabe or wrong exploit code was sent.

D) Victim receives and executes exploit code but no worm code is downloaded.

E) Victim is successfully infected

8/11 16:35 UTC external worm outbreak
8/11 **17:42** UTC first infection (from outside)
8/11 **18:20** UTC first infection (from inside)
8/12 6:50 UTC internal (massive) outbreak

Blaster's Infection Summary

Results of Blaster observation (8/11 16:35 – 8/16 0:20 UTC):
- 73 distinct attackers, whereas
- only **215** successful infections observed

→ **almost not worm code (≠exploit code) transmitted over backbone**

Reasons:
- multi-stage nature of Blaster (various protocols; WAN delay/congestion)
- preference for local scanning

- 47 victims (in 13 adjacent /16 networks) infected by most successful host
- 11 out of top 21 most successful hosts belong same /16 network
- 3 days after outbreak new infection activity peak (stage C)

→ **slow patching procedures of hosts visible**

Other findings:
- top ten most successful attackers infected 138 (64%) of the victims
- 76% infections originate from inside
- 24% infections originate form outside

Agenda

1) Introduction

2) Flow-Level Backbone Traffic

3) Network Worm Blaster.A

→ 4) E-Mail Worm Sobig.F

5) Conclusions and Outlook

E-Mail Worm Sobig.F

An e-mail in my INBOX:

Would you open this attachment to watch the screensaver?

Sobig.F:

- Sobig.F outbreak on Aug 19th, 2003, before 10:00 UTC
- worm is in attachment
- uses own SMTP engine to send itself to recipients found in local files
- (unsuccessful) update feature (blocked by timely server shutdowns)

Testbed for Sobig.F

Sobig.F network measurements

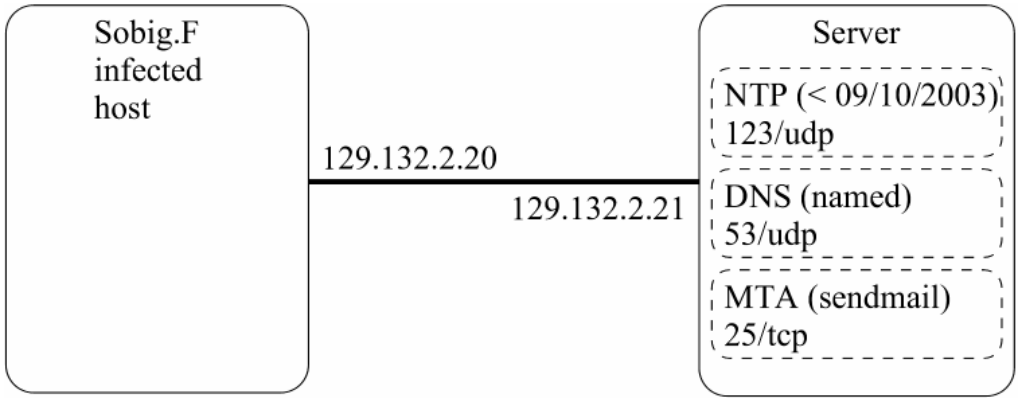
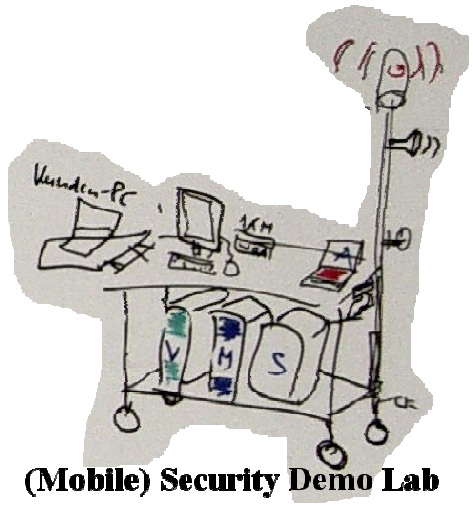
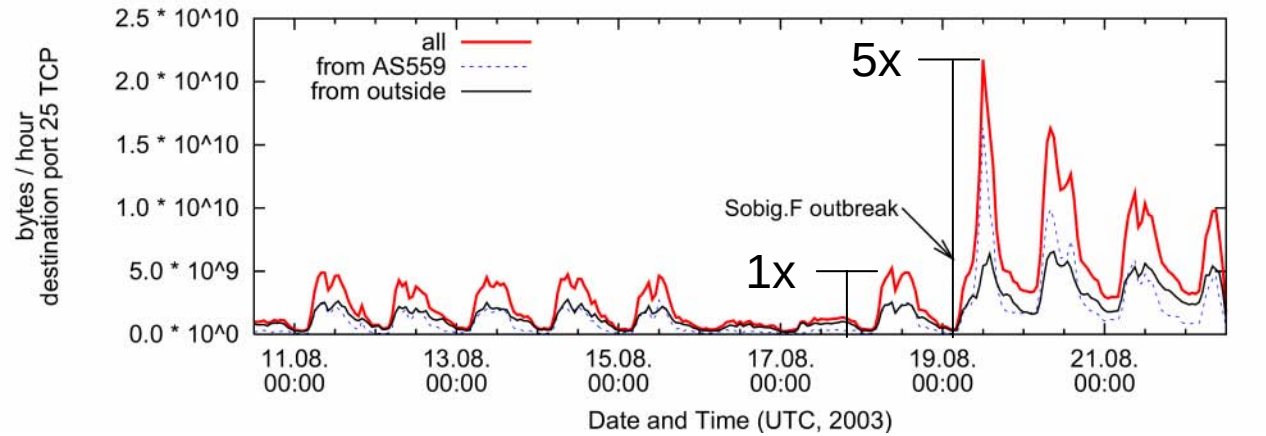


Figure: Tesbed for Sobig.F worm



References: Mobile Security Demo Lab mSDL
http://www.csg.ethz.ch/research/projects/mSDL

Sobig.F flow based analysis:

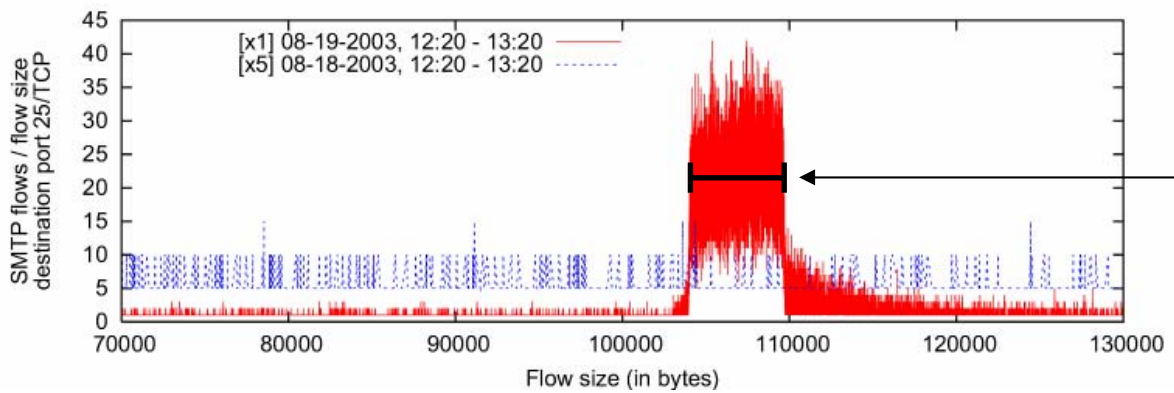


Almost **fivefold increase** in e-mail traffic (bytes/hour) during initial spreading of Sobig.F. Outbreak.

Figure Sobig.F worm: **SMTP traffic volume per hour**

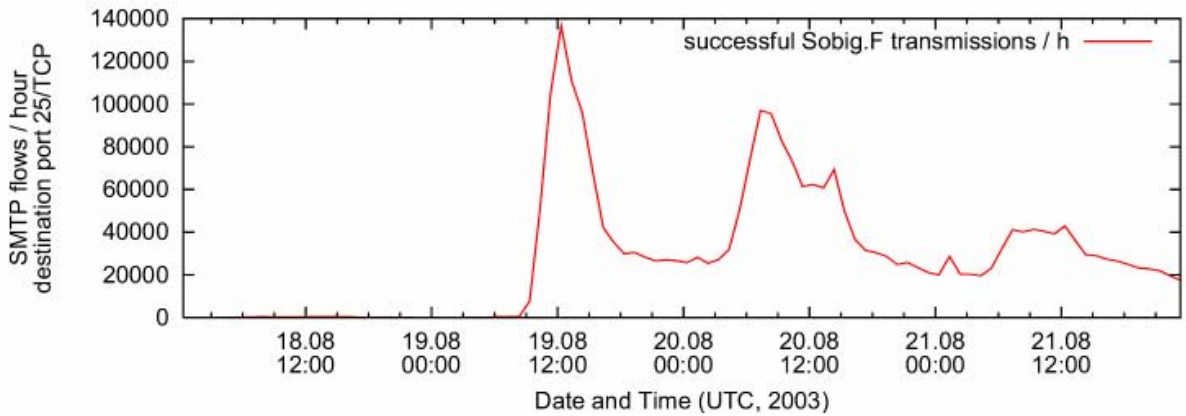
E-Mail Worm Sobig.F

Sobig.F flow based analysis:



Significant raise in the number of SMTP flows in the range of **103 kB – 125 kB** during outbreak; many *TCP caused packet retransmissions*

Figure: SMTP flow size distribution before and during Sobig.F



Up to **140‘000** Sobig.F e-mails per hour transmitted into or out of AS559 during peak of worm outbreak

Figure: Number of Sobig.F transmissions per hour

Conclusions

Results:

- **spreading** event of massive worms clearly **visible on flow-level in backbone**; forensics on flow-level backbone traffic possible
- **delay** in the order of hours between external and AS559-internal outbreaks → early detection and prevention in backbone and access networks could reduce worm impacts drastically

Blaster.A (multi-stage network worm):

- short network test of *Blaster pre-release* detected
- significant changes of various traffic paramters during outbreak
- backscatter effects due to non-existent hosts (ICMP)
- *ineffectiveness of* certain temporary port blocking *countermeasures*
- *low frequency of actual worm code transmissions* (due to Blaster‘s multi-stage nature and preference for local scanning)

Sobig.F (email worm):

- many TCP packet retransmissions due to greedy spreading algorithm

Outlook

- continuation of long-term **capturing** efforts (DDosVax NetFlow archive)

- further **analyses** of massive worms and Internet attacks planned

- development of **algorithms for early worm outbreak detection** (some already published at IEEE WETICE 2005: Host behaviour based worm detection; Entropy based worm detection)

- contributions for an **Internet attack detection system** for backbone operators based on flow-level traffic (our „UPFrame“ system)

Thanks for your attention!

Any questions?

You can reach Thomas Dübendorfer at:
duebendorfer@tik.ee.ethz.ch

The DDoSVax project at ETH (publications):
http://www.tik.ee.ethz.ch/~ddosvax/