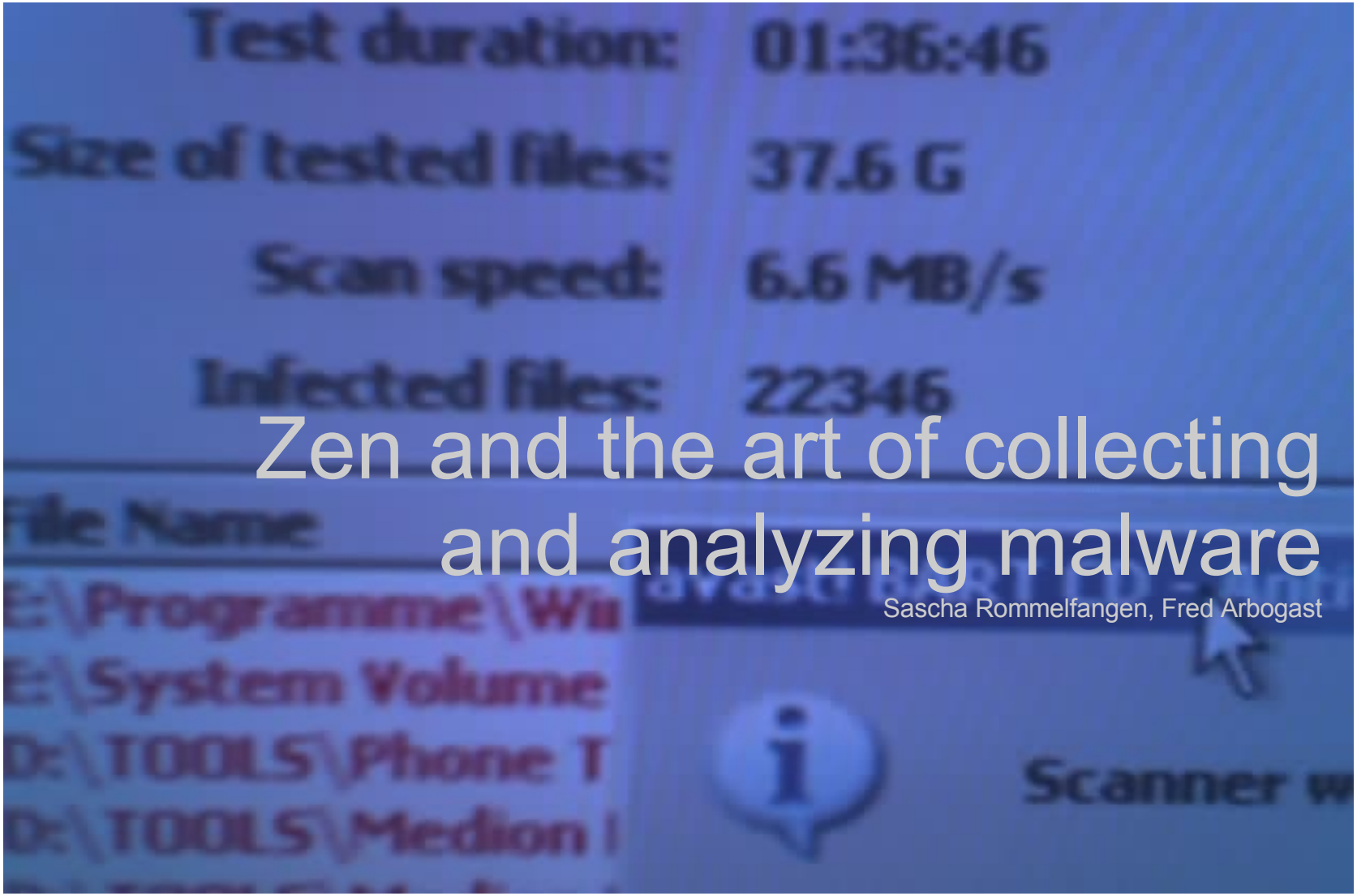
CanSecWest/core06



Zen and the art of collecting and analyzing malware

Sascha Rommelfangen, Fred Arbogast

SES ASTRA
An SES GLOBAL Company

Zen and the art of collecting and analyzing malware

Computer Security Research & Response Team

CSRRT-LU

Outline

- Introduction
- Setup to collect malware
- Statistics
- Analysis
- Live Demo
- Future development
  - early warning/reacting system approaches
  - interactive malware database
- Conclusion

SES ASTRA
An SES GLOBAL Company

Zen and the art of collecting and analyzing malware

Computer Security Research & Response Team
CSRRT-LU

Definition of malware

- Umbrella term for malicious software
- Not to be confused with *defective software*
- Designed to infiltrate, damage, control or abuse computer systems without owner's consent
- Legal vocabulary: computer contaminant
- Also used: scumware
- Worms, virii, root kits, spyware, adware

The tools used

- mwcollect by Georg Wicherski
  - (http://www.mwcollect.org)
- Nepenthes by nepenthes team
  - (http://nepenthes.sourceforge.net)
- Focus on nepenthes as mwcollect has merged with nepenthes
- Joint effort will result in a more powerful tool

Things both tools have in common

- “Low interaction” honeypots
- passive
- catching autonomously spreading malware
- Running in non-native environments
- simulating network services
  - mwcollect: vulnerable built-in services
  - nepenthes: additionally 'pre-infected' services
- acting upon exploitation attempts
  - Downloading malware
- Both tools are Free and Open Source software

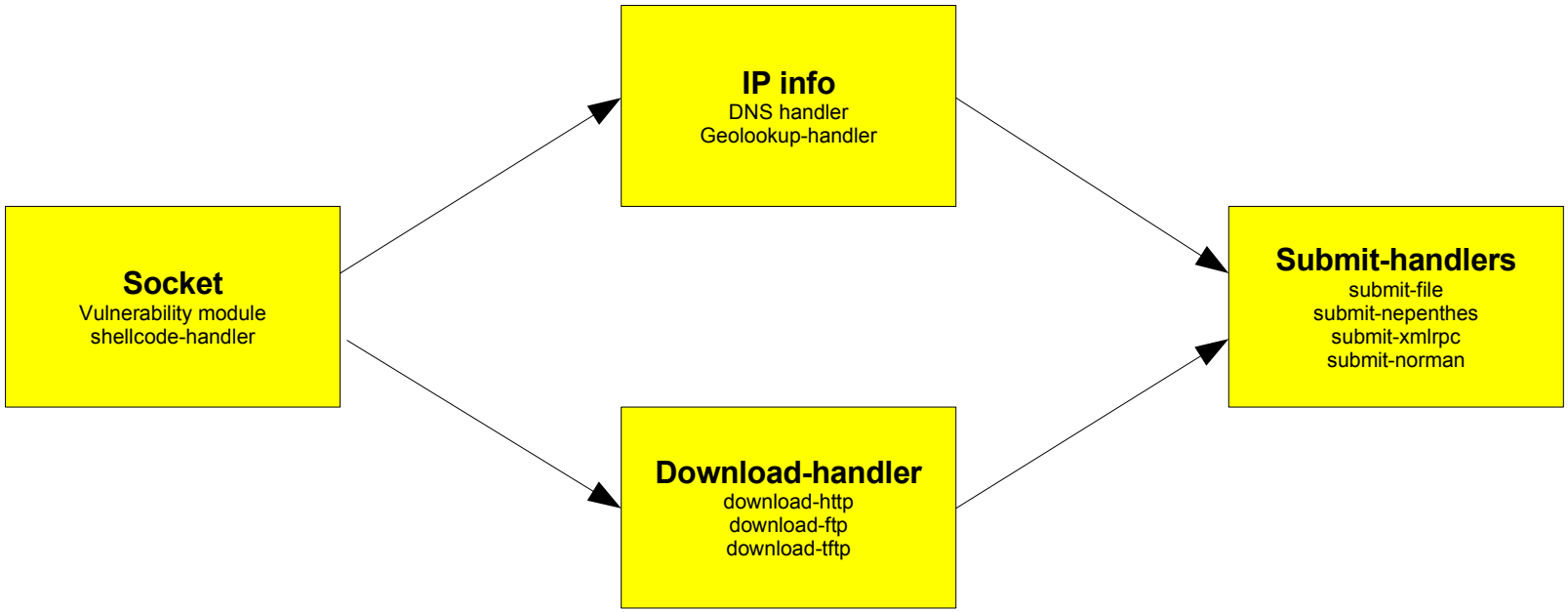
Tools - nepenthes

- Emulates native and non-native vulnerabilities
- Modular
  - Know a new exploit, add it as a module
- Support for geolocation information
- Support for submitting malware and additional information
  - Other instances of nepenthes (distributed installation)
  - XML-RPC

Vulnerabilities

- 'native' vulnerabilities:
  - RPC-DCOM (135, 139, 445, 593)
  - LSASS (445)
  - WINS (42)
  - MSSQL (1434)
  - ASN.1 library in IIS, SMB (80 and 445)
  - IIS (443)
  - NetDDE (139)
  - Message queueing (2103, 2105, 2107)
  - UPNP (5000)
- 3rd party vulnerabilities:
  - Kuang2 (17300)
  - Mydoom (3127)
  - Bagle (2745)
  - sasser_ftp (5554, 1023)
  - Sub7 (27374)

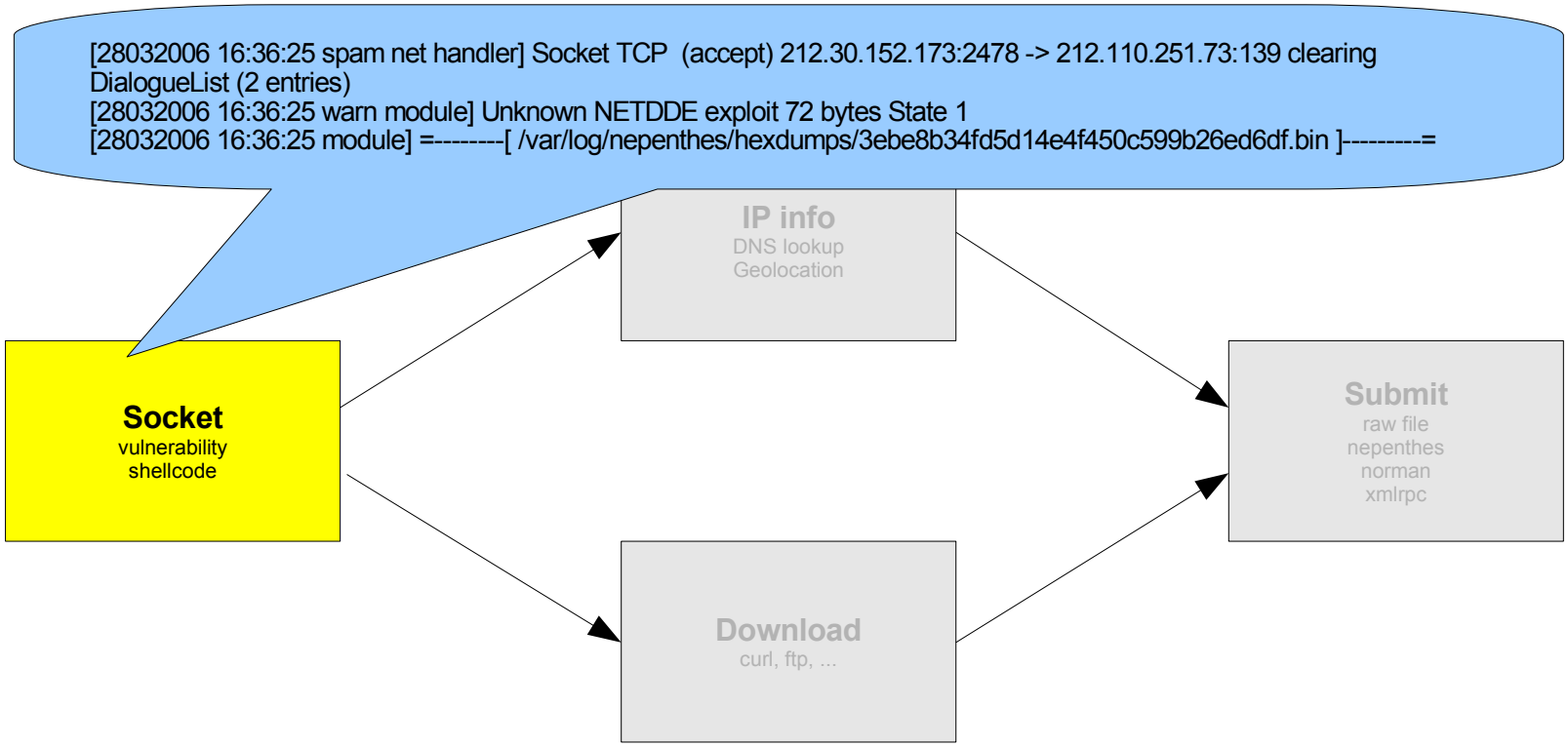
Nepenthes information flow - modules/handlers



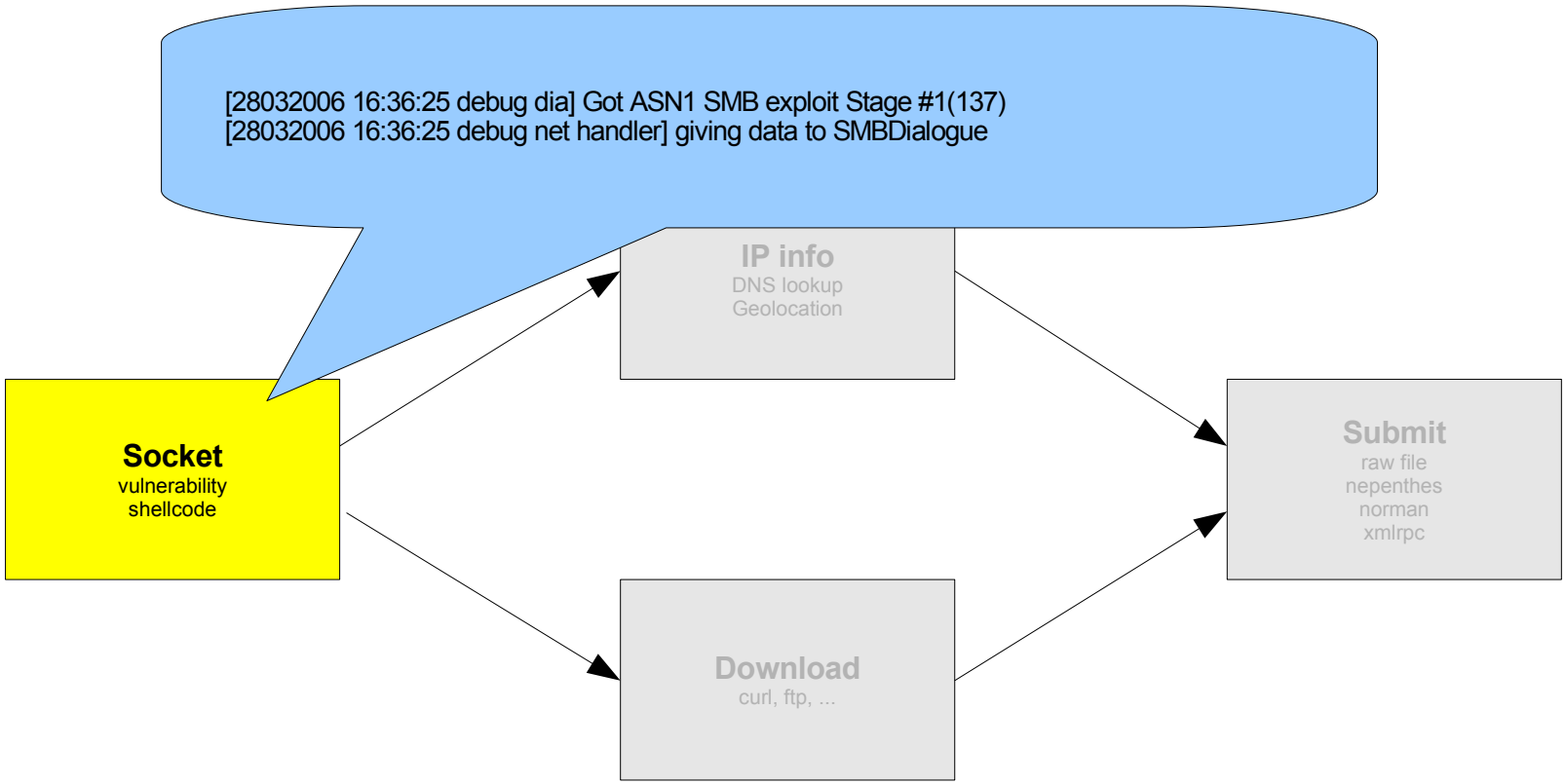
Categories of modules/handlers (1)

- Vulnerability module
  - Different modules for simulating the vulnerabilities
- Shellcode-handler
  - Per shellcode one module
  - Common Shellcode Naming Initiative

Nepenthes information flow



Nepenthes information flow



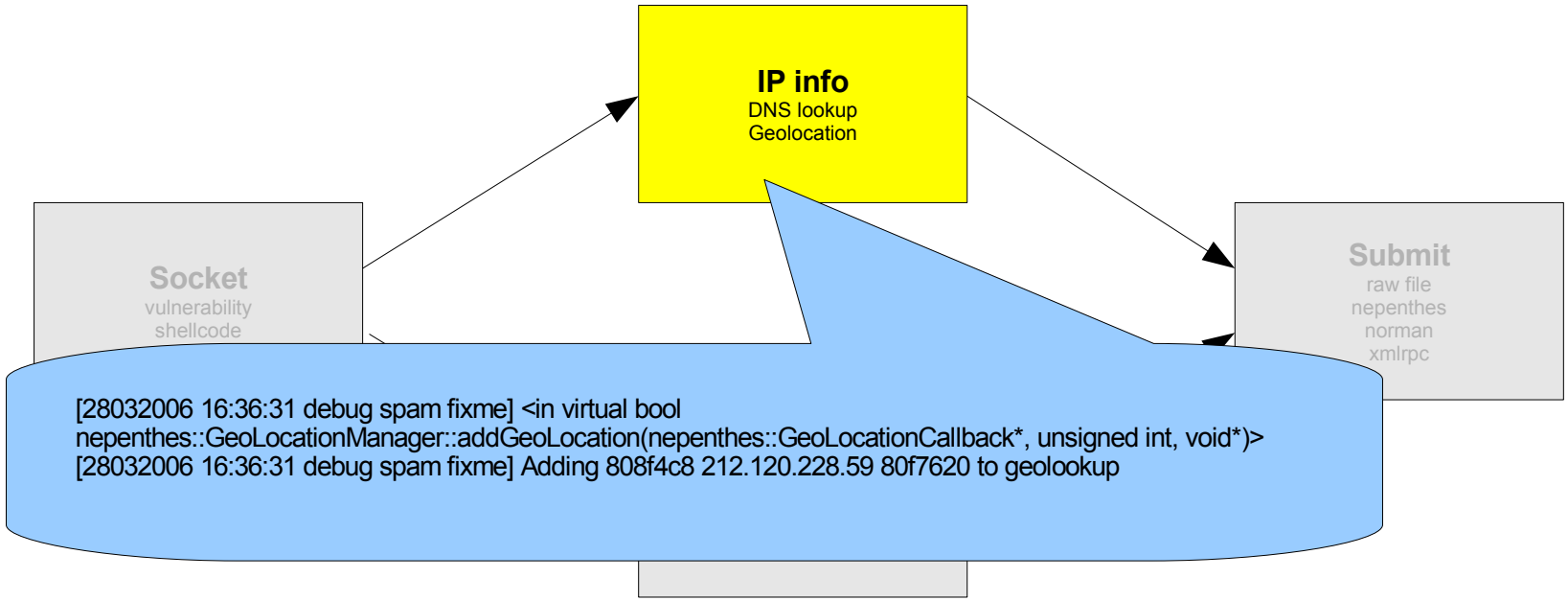
Categories of modules/handlers (2)

- Geolocation-handler (some alternatives)
  - Resolves IP address to location information
- DNS-handler
  - Delivers resolved domain name
- Download-handler
  - Downloads through curl
    - Provides http and ftp protocol
  - Download ftp
    - Needed as curl is not the same than the messy M$ client
    - Netcat is doing the job

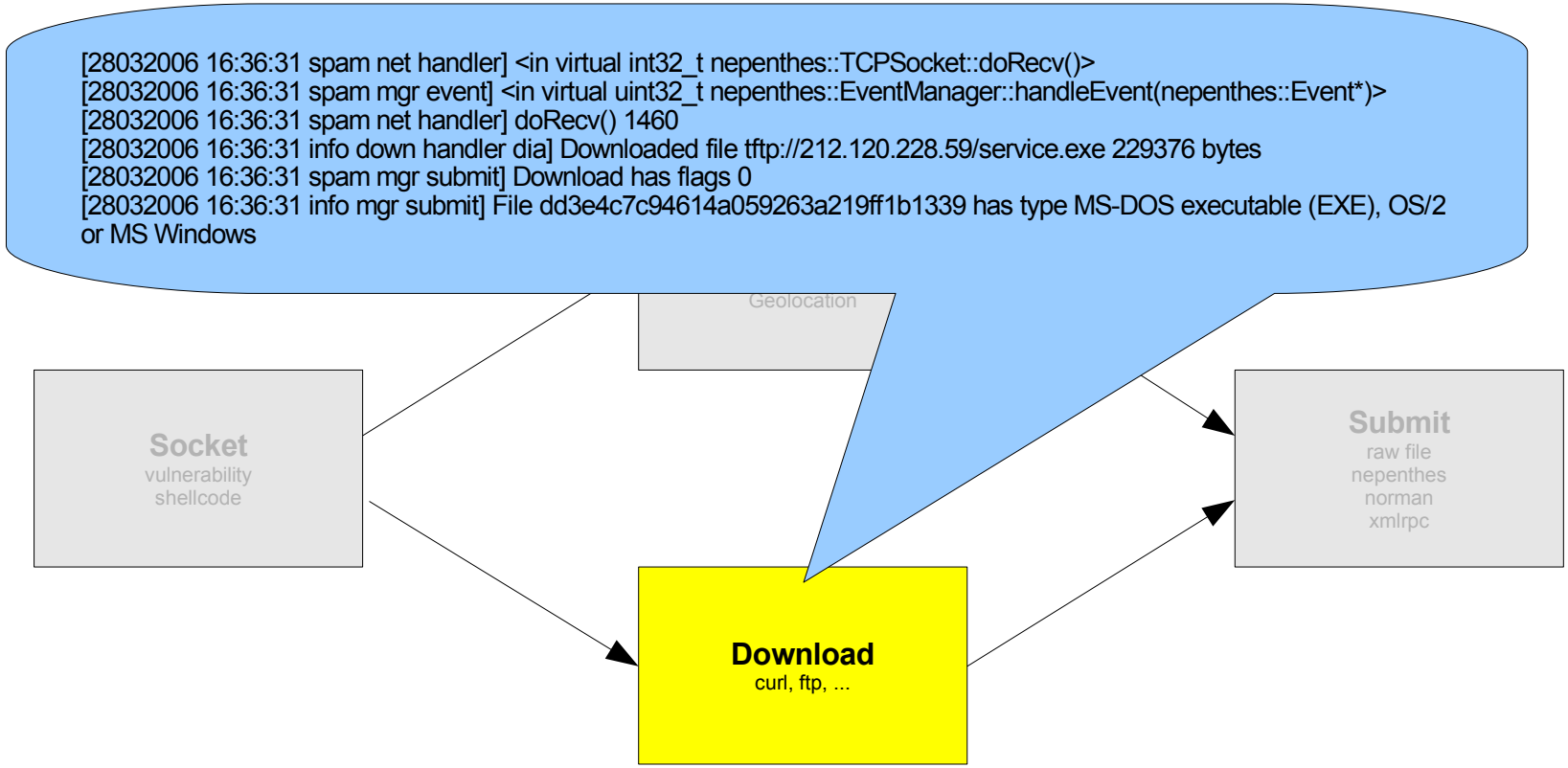
Categories of modules/handlers (3)

- Download-handler cont'd
  - Download tftp
    - Support for tftp protocol
    - Max filesize 4MB
    - Can not handle DNS for the moment
  - Download nepenthes
    - Listens for file transfers from other nepenthes agents
    - Port can be set in the config file
    - transfer is simple and bandwidth optimised

Nepenthes information flow



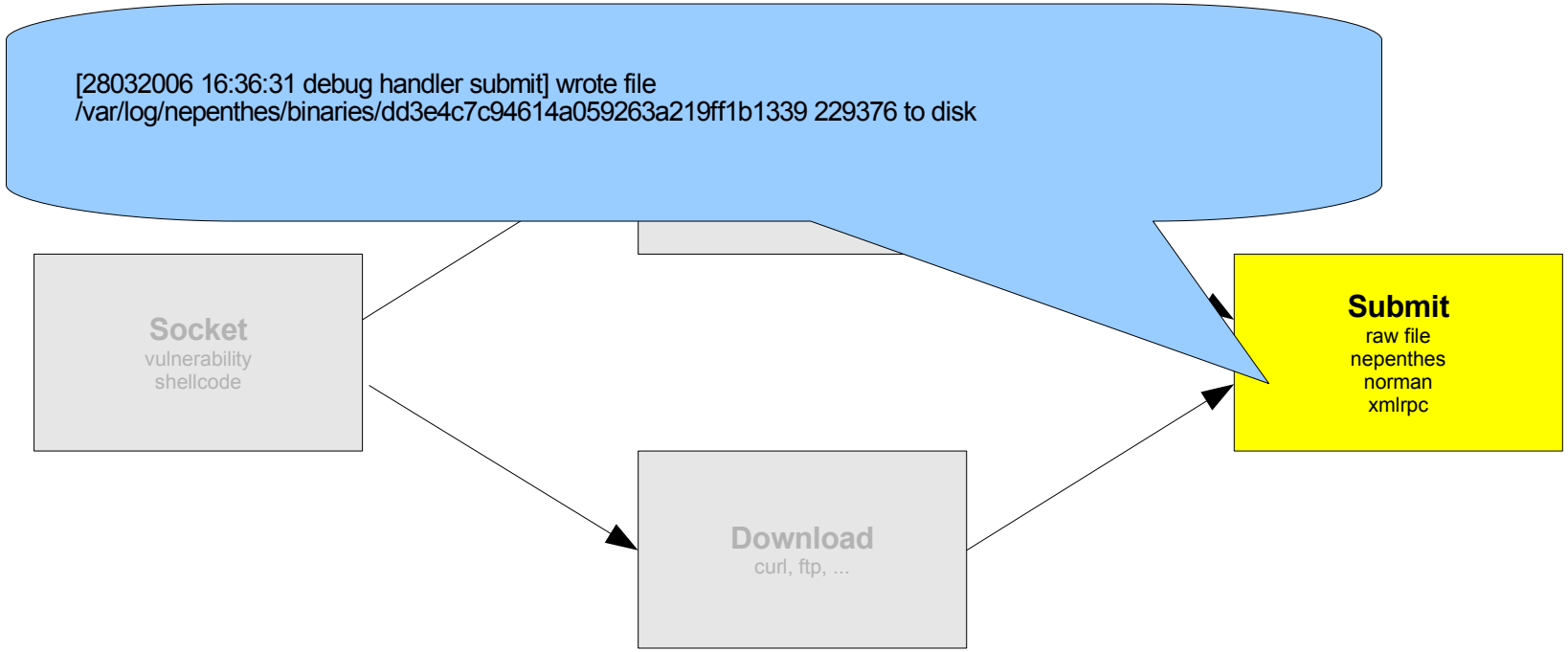
Nepenthes information flow



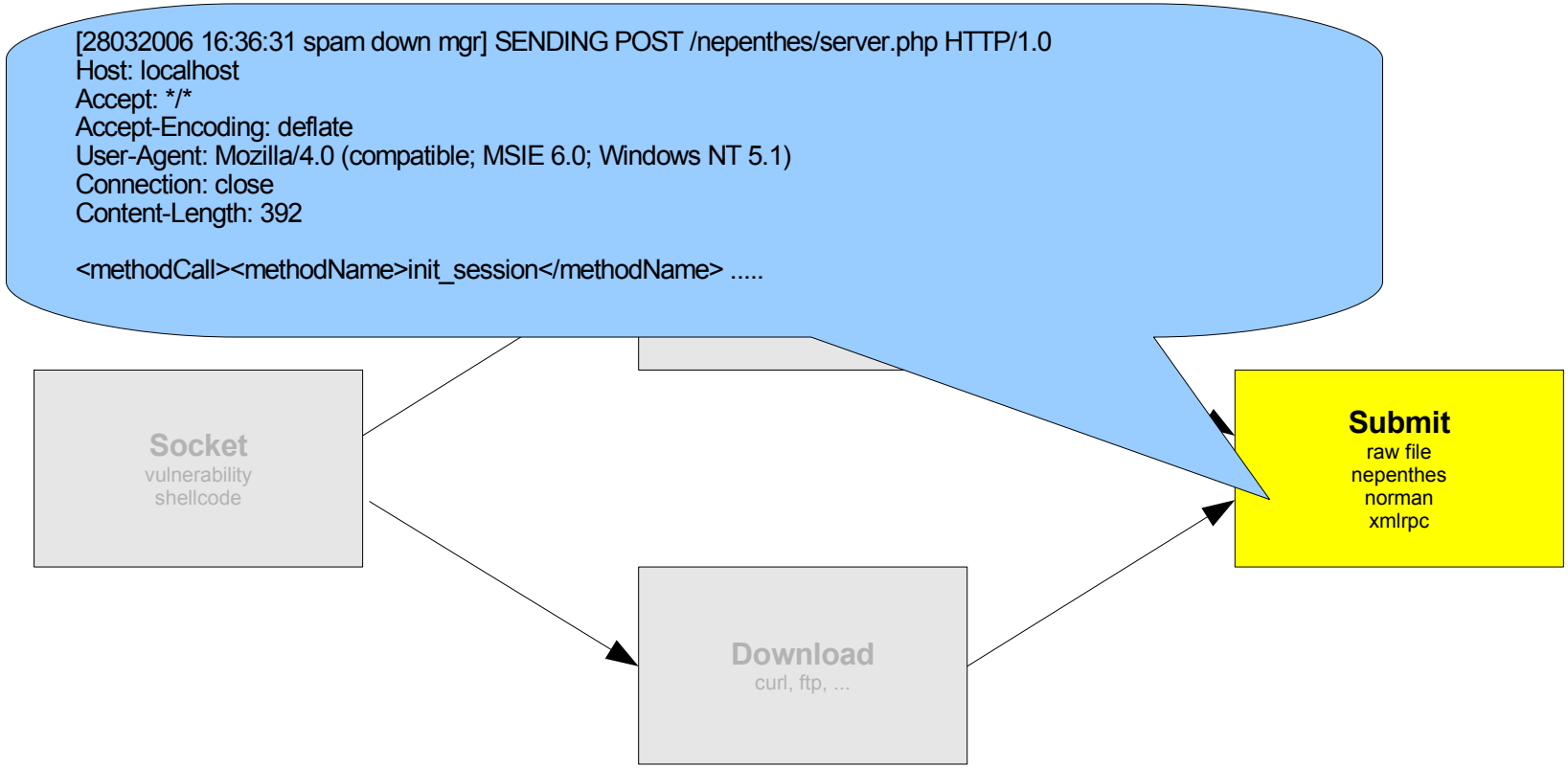
Categories of modules/handlers (4)

- Submit-handlers
  - Submit-file
    - Dumps to a file on HDD
  - submit-nepenthes
    - Submits information to a central server
    - Currently receiving from Telecom Italia Early Warning Team
  - Submit-norman
    - Submits file to norman sandbox
  - Submit XML-RPC
    - Submits information to applications outside nepenthes

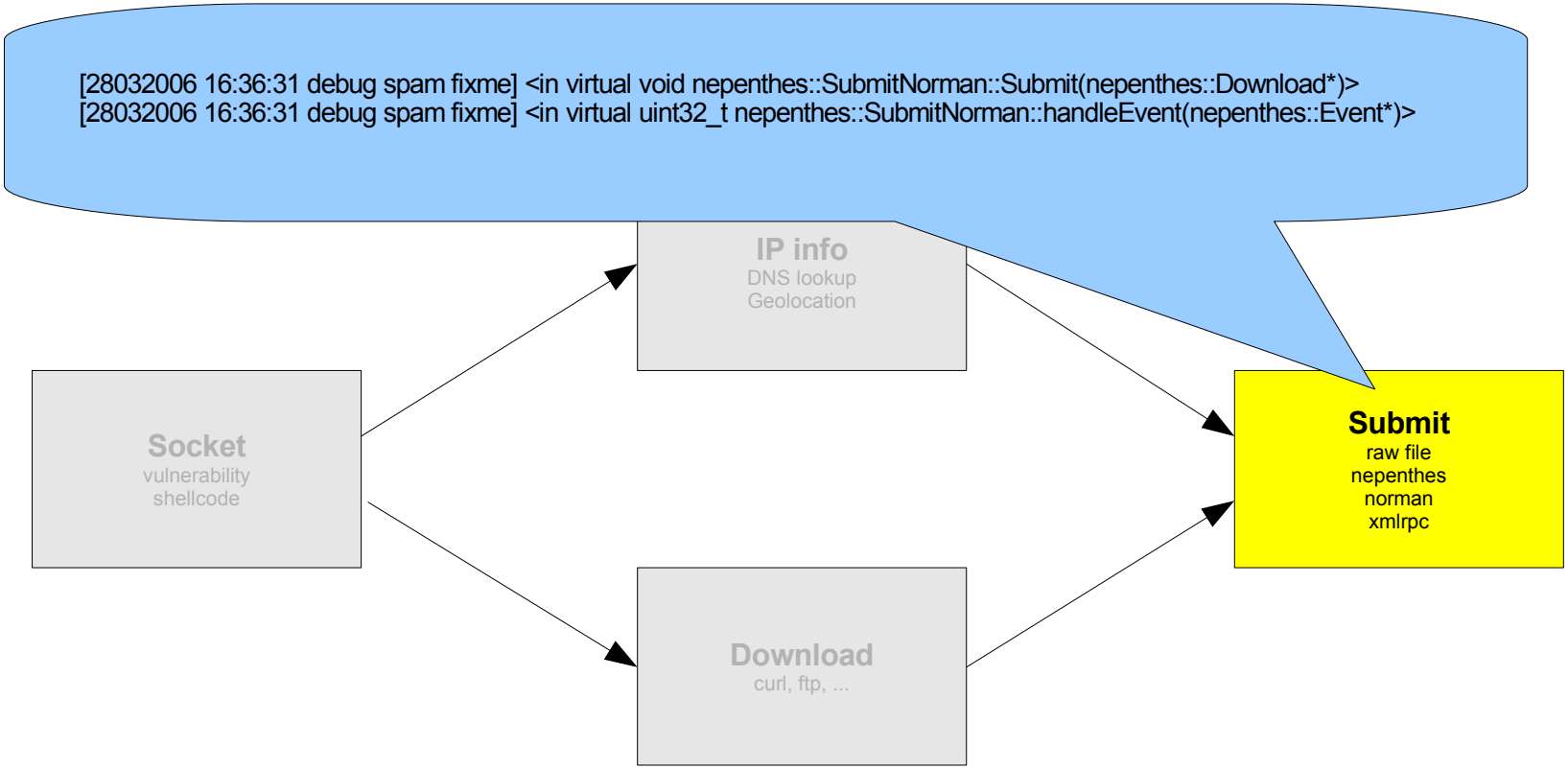
Nepenthes information flow



Nepenthes information flow



Nepenthes information flow



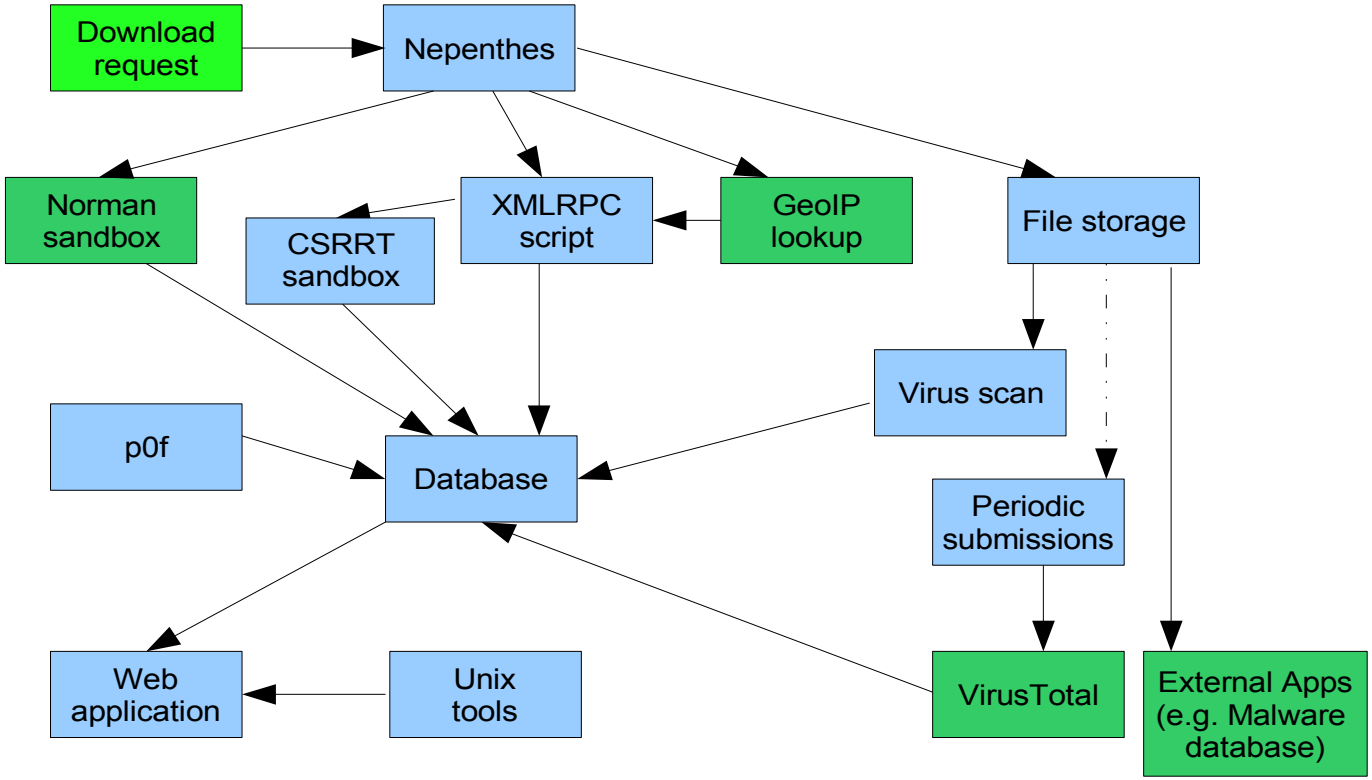
Additional information collected

- Extension to nepenthes - stored in database
  - Platform information (p0f-sql)
    - P0f hack to submit information into DB
  - 4 AV product results from local machine
    - Extendable
    - Signatures hourly updated
  - 24 AV results from VirusTotal (added later)
  - 2 sandbox results
    - Submitted to http://sandbox.norman.no
    - Submitted to our own POC sandbox (added later)

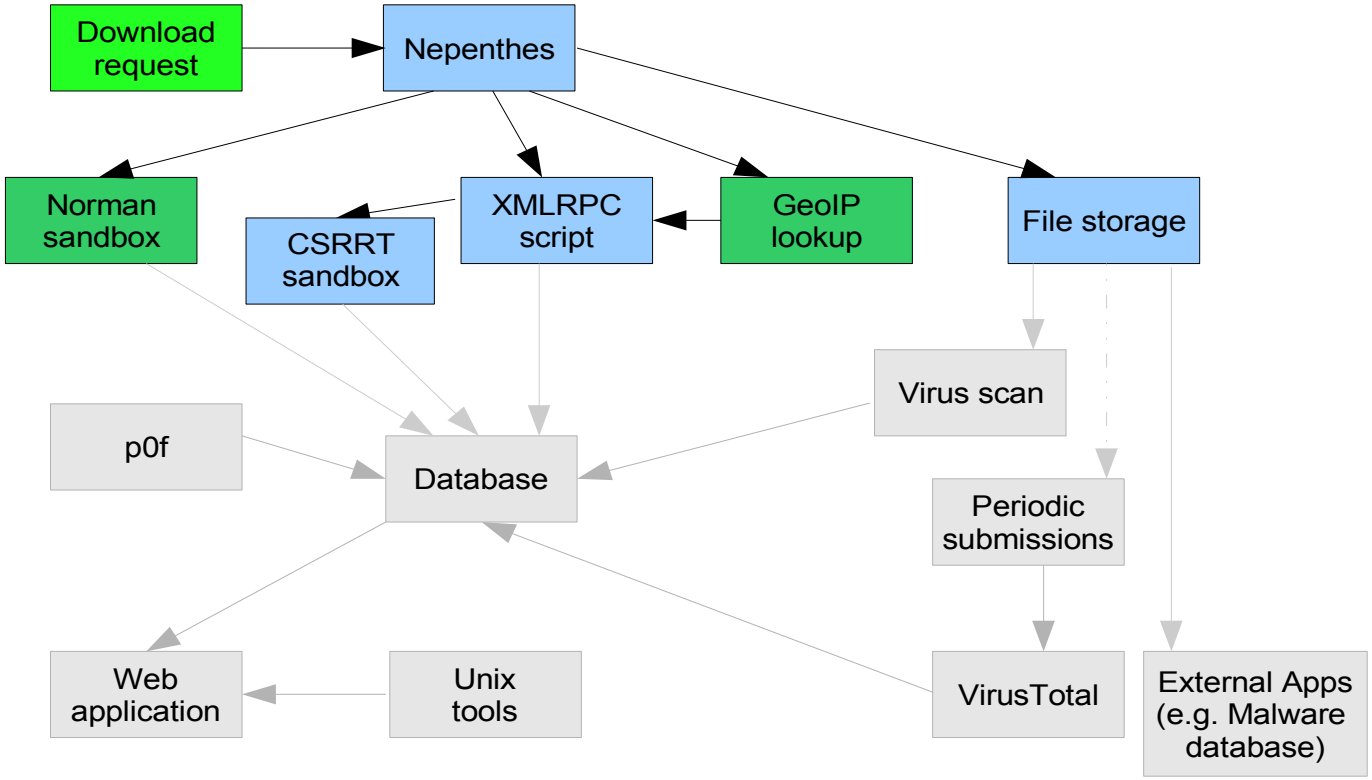
Full information set collected

- Various static analysis
  - file, upx, hexdump, strings, objdump
- Number of hits
- First/last seen
- Number/names of recognized virii
- Sandbox results
- Hex-dump of file (browseable)
- IP/URL from where fetched
- System
- Latitude, Longitude, Country, City

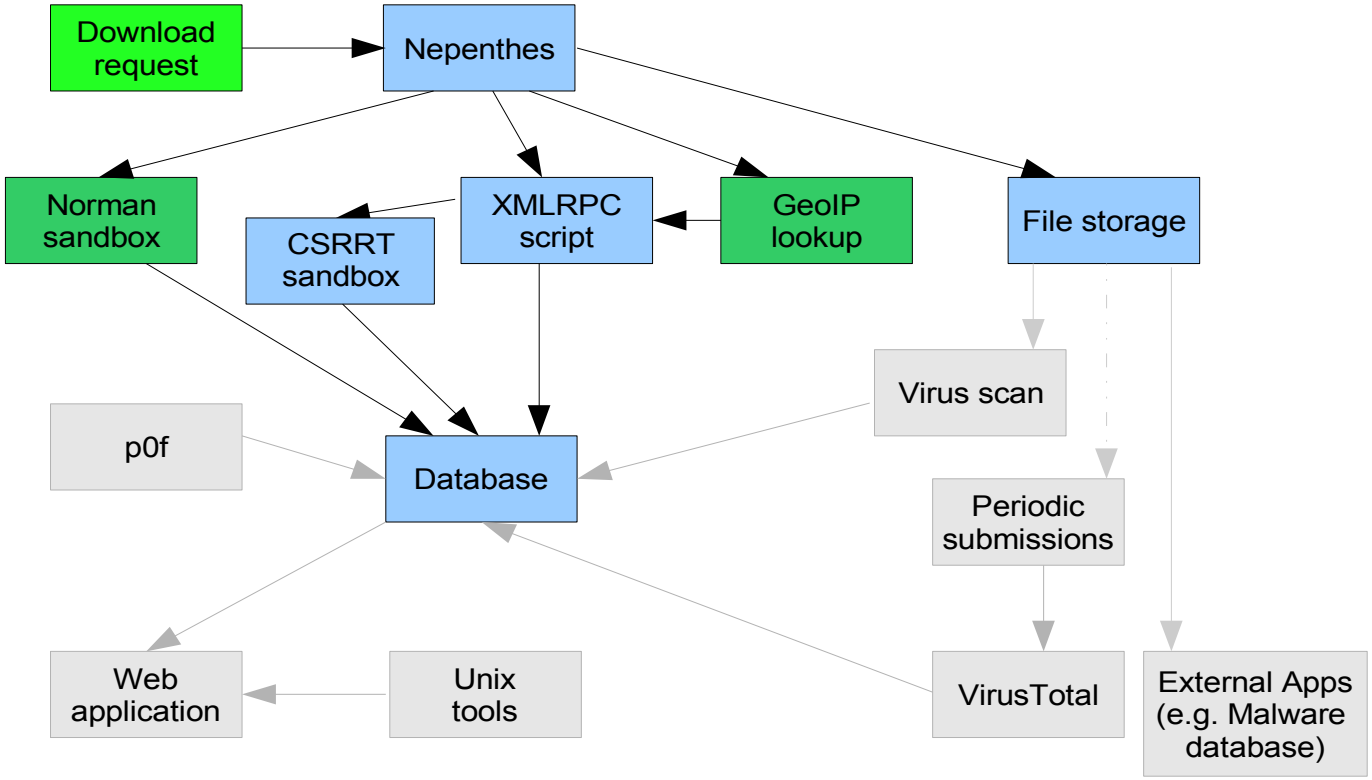
Setup to collect malware – flow



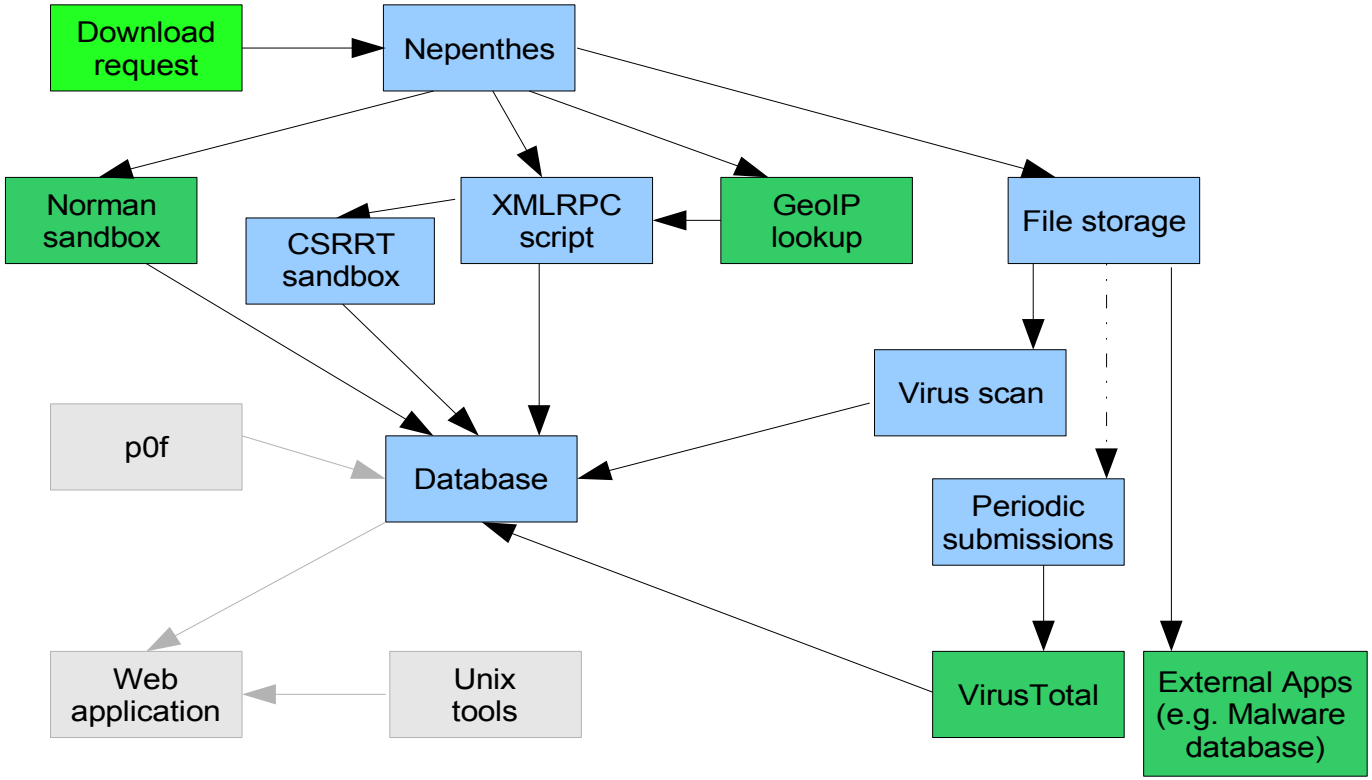
Setup to collect malware – flow



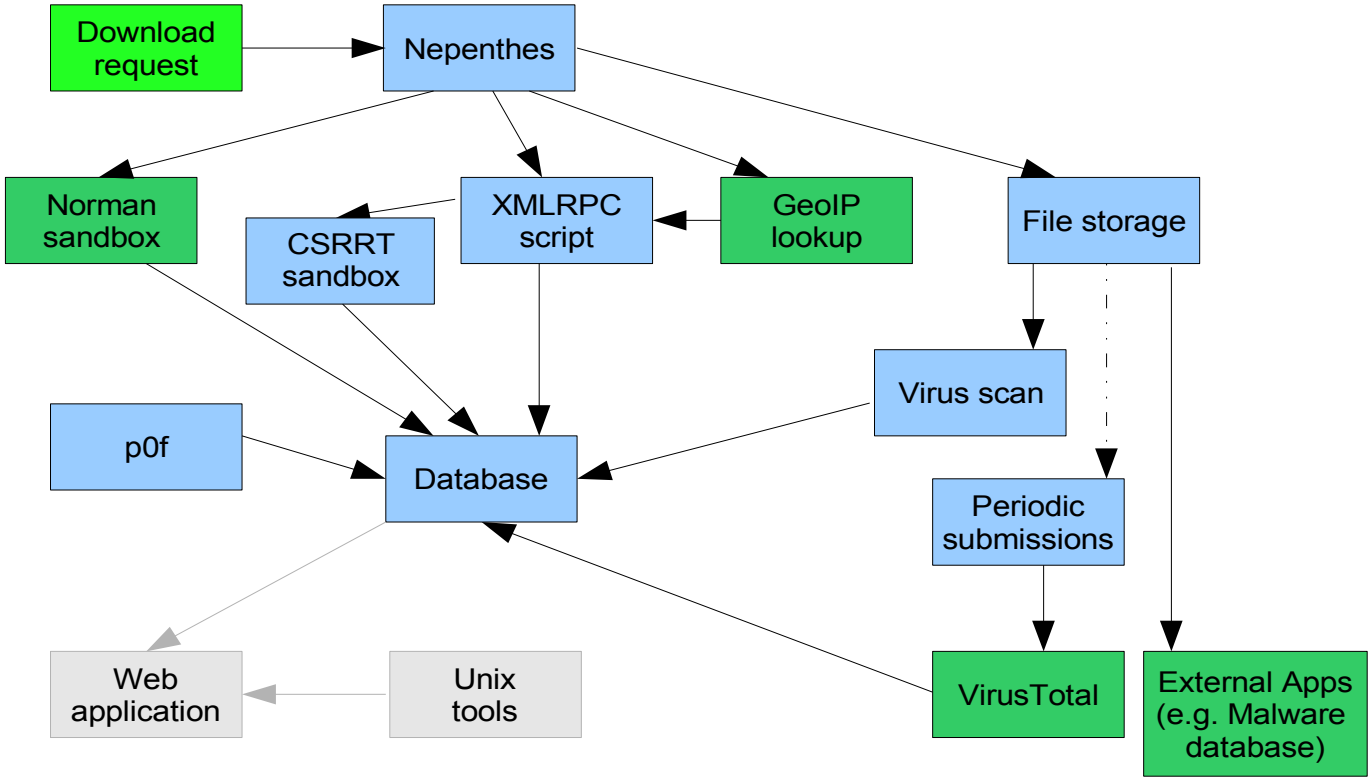
Setup to collect malware – flow



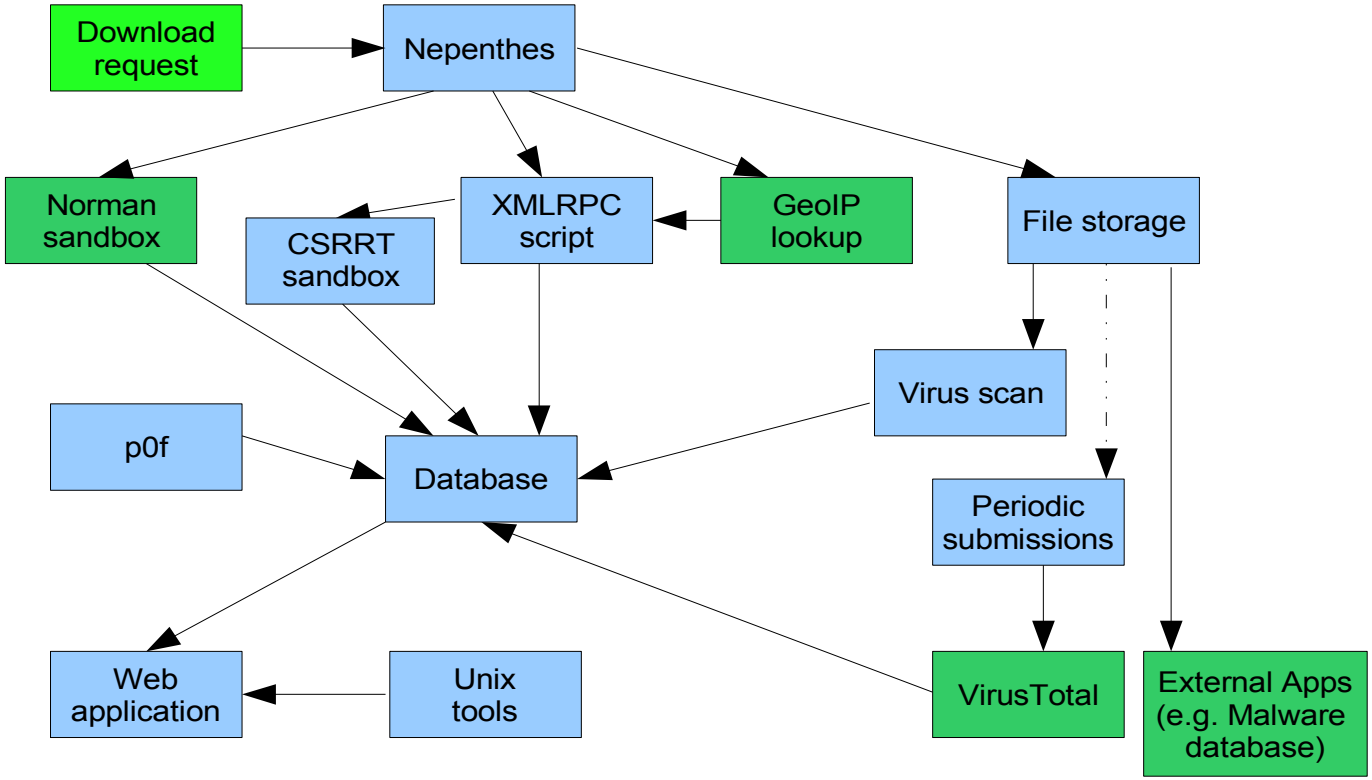
Setup to collect malware – flow



Setup to collect malware – flow



Setup to collect malware – flow



Statistics

- There are three kinds of lies: lies, damned lies, and statistics. Benjamin Disraeli (1804 - 1881)
- 1st set, collected with mwcollect:
  - Approx 600,000 files (9.2 GB)
  - 542 unique (80 MB)
  - 529 executables
  - File length: 100 to 1,145,856 Bytes
  - Time frame: 6 weeks (April - June 2005)
  - 503 MS-Windows executables
  - 26 MS DOS executables

Statistics

- 1st set continued
  - 52% of the files were detected by all 4 virus scanners
  - 17% of the files were detected only by 3 virus scanners
  - 25% of the files were detected only by 2 virus scanners
  - 3% of the files were detected only by 1 virus scanner
  - 2% were defective
- When scanning files later -> some files detected as Zotob
  - During collecting time there was no Zotob signature!
    - false positive?
    - test-run?

Statistics

- 2nd set, collected with nepenthes:
  - 2,079 unique files
  - 209,327 malware downloads complete
  - 13% using anti debug/emulation techniques
  - 1,852 MS-Windows executables
  - 227 MS-DOS executables
  - File length: 1,024 – 1,323,222 (1.3MB) bytes
  - Time frame: December 2005 – March 2006

Statistics

- Result of immediate scan:
  - Results of virus scan, directly after reception with up-to-date signatures:
    - 69.5% Norman Sandbox
    - 68.5% Bitdefender
    - 58.0% Antivir
    - 49.5% F-Prot
    - 31.8% ClamAV
  - Are signature based systems really future-proof?

Statistics

- Results of re-scan:
  - 96.1% Panda
  - 91.2% Norman
  - 85.9% Antivir
  - 85.9% Avira
  - 85.1% Kaspersky
  - 84.7% DrWeb
  - 84.5% Fortinet
  - 83.9% McAfee
  - 83.8% BitDefender
  - 80.4% VBA32
  - 80.1% CAT-QuickHeal
  - 79.8% NOD32v2
  - 78.9% UNA
  - 77.2% AVG
  - 76.3% Symantec
  - 75.7% Ewido
  - 72.4% F-Prot
  - 65.9% Sophos
  - 65.1% TheHacker
  - 64.1% Ikarus
  - 57.2% eTrust-Inoculate
  - 54.3% Avast
  - 50.7% ClamAV

SES ASTRA
An SES GLOBAL Company

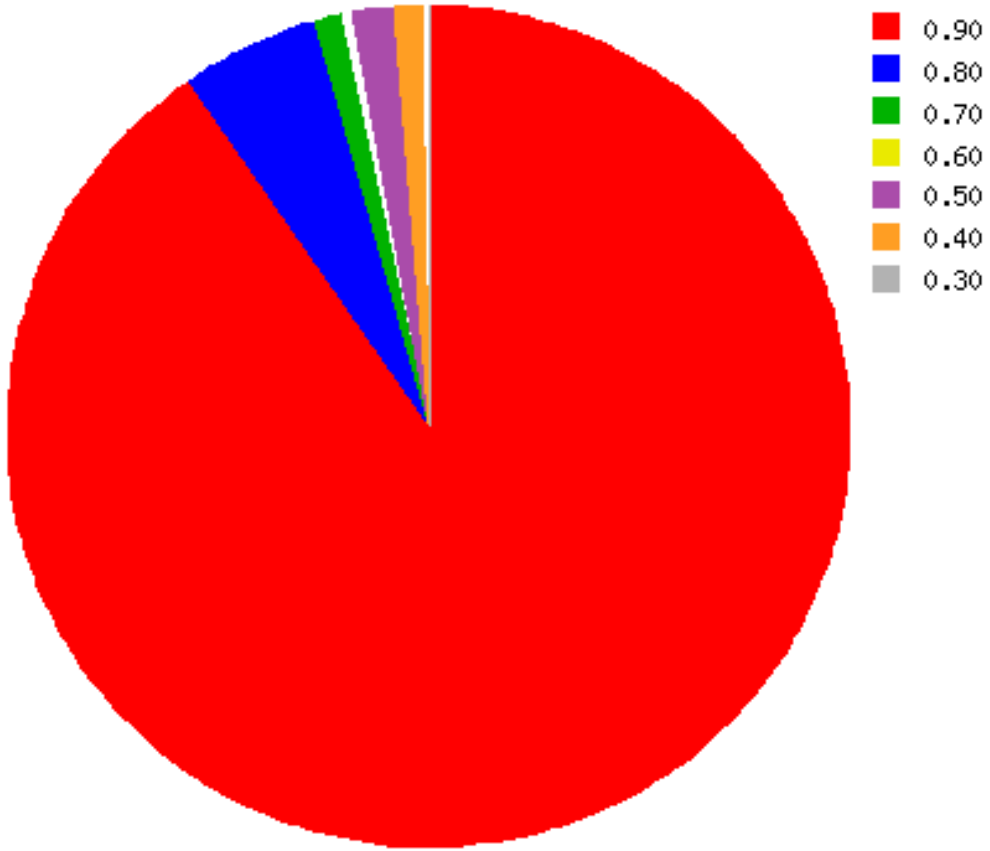
Zen and the art of collecting and analyzing malware

Computer Security Research & Response Team

CSRRT-LU

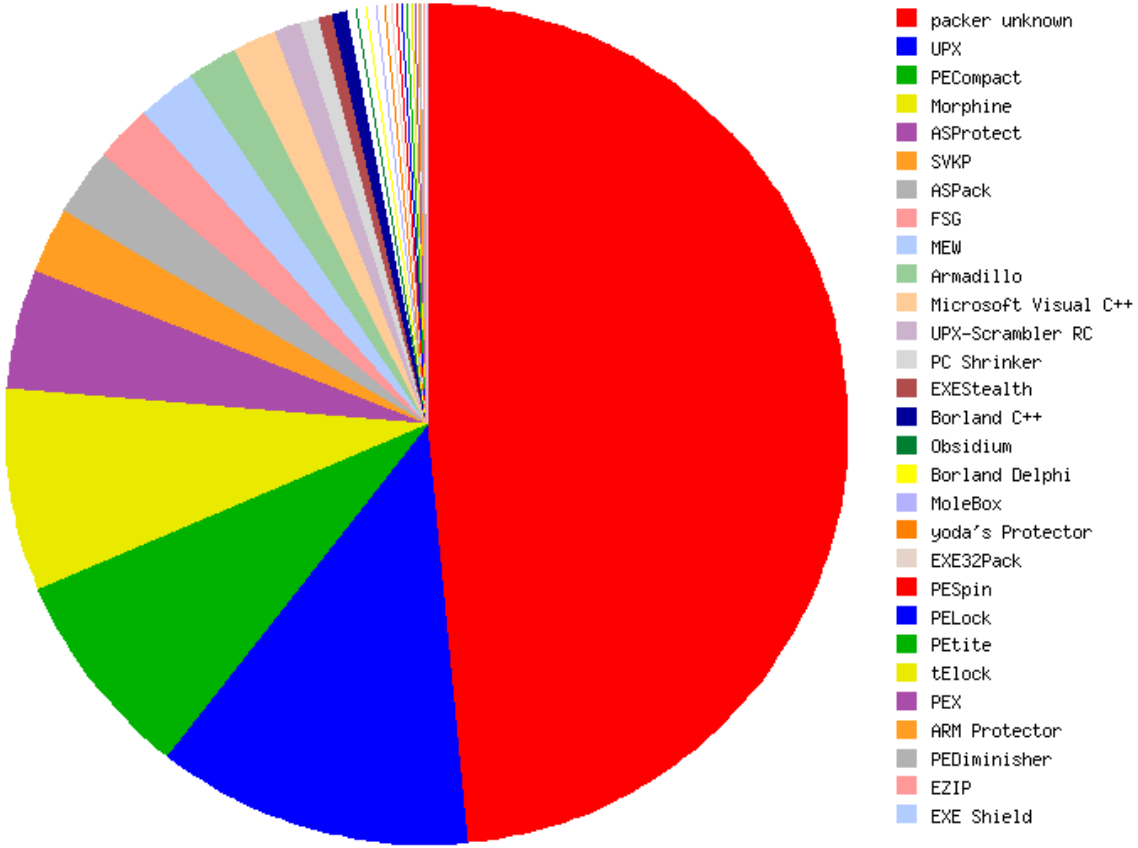
Statistics

- Packing/Encrypting statistics using bzip2



Statistics

- Packing/Encrypting statistics:



SES ASTRA
An SES GLOBAL Company

Analyzing Malware - Side-effects

- Malware hides from the analyzer and obfuscates its techniques
- Automated processes not 100% reliable
  - Anti-virus products, current sandbox techniques
- Last resort: manual investigation
  - Disassembler, Debugger, file monitors, registry monitors, Virtual Machines
  - Very time consuming and/or requires high skills

Ways to fool the analyzer

- Modified binary
  - (multiple) Packing
  - Encrypting
  - Header crippling
- Test presence of Debugger/Disassembler
  - SoftICE, OllyDbg, Breakpoints, Vmware, ...
  - http://www.honeynet.org/papers/bots/botnet-code.html
- Usage of file droppers
  - Dropper downloads malware and executes it
  - Malware makes usage of other malware already downloaded (e.g. browser hijacker vmmon32.exe)

Automated analysis

- Virus Total:
  - Free service scanning files with 24 AV products
  - Submits by default samples to AV vendors
  - Automated submission through extensions
  - Virus Total sends back mail with report
  - Most of the time at least one AV product finds malware
  - Cooperativeness to extend results (e.g. XML, more details, ...)
  - Negative point:
    - Slow – agreed on a 60s interval when sending all files (adding more resources in the future)

Automated analysis

- Norman sandbox:
  - APIs simulating a Windows Computer
  - Some of the APIs simulate the Network/Internet connectivity
  - Automated submission through nepenthes
  - Sandbox sends back mail with report
  - Negative points:
    - often not working because of filled up mail queue
      - Necessity to resubmit
    - Often trapped into anti-debug code
  - Have to trust the output!

Norman Output

Googlesetup.exe : [SANDBOX] contains a security risk - W32/Spybot.gen3 (Signature: W32/Spybot.AHWZ)
[General information]
* **Locates window "NULL [class mIRC]" on desktop.
* File length: 133120 bytes.
* MD5 hash: df2eaaf757053a4a0209c4668efd8d1c.

[Changes to filesystem]
* Creates file C:\WINDOWS\SYSTEM32\Googlesetup.exe.
* Deletes file 1.

[Changes to registry]
* Creates value "Google service"="Googlesetup.exe" in key "HKLM\Software\Microsoft\Windows\CurrentVersion\Run".
[...]

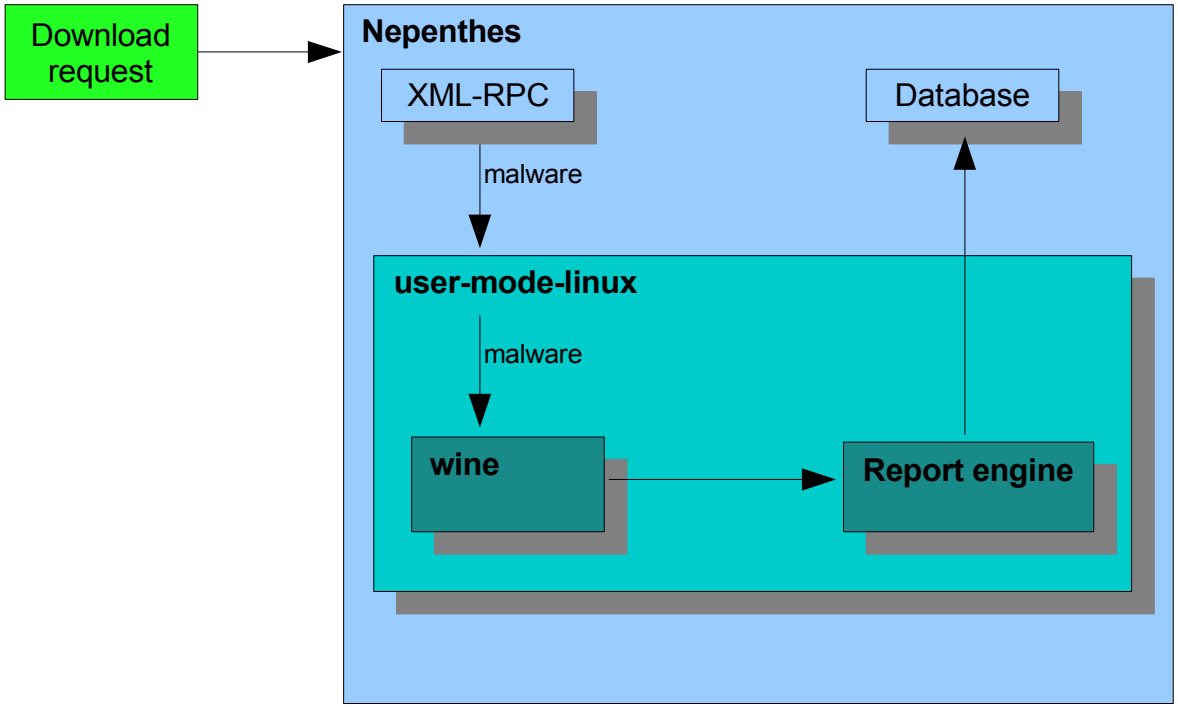
[Network services]
* Looks for an Internet connection.
* Connects to "der.ifconfig.us" on port 7000 (TCP).
* Connects to IRC Server.

[Signature Scanning]
* C:\WINDOWS\SYSTEM32\Googlesetup.exe (133120 bytes) : W32/Spybot.AHWZ.

Automated analysis using wine (1)

- wine as a (cheap) sandbox approach
  - Why?
    - Signatures suck
    - wine executed 72% out of 2199 malware files
  - How?
    - Compare .wine directory with an unmodified one
    - Use debug and trace messages from wine
    - Create report from what is known (~signature)
  - Security?
    - Outbreak is possible - include an assembler program that executes linux system calls via int 80h in the .text section of the windows executable
    - we're using user-mode-linux

Automated analysis using wine (2)



Live Demonstration

- http://nepenthes.csrrt.org:10080/nepenthes/

Developments and future steps (1)

- Early warning / reacting system (so far implemented)
  - Monitoring and visualization of outbreak waves
  - Live export of most common attacker IP list
    - To be imported into Firewalls, IDS, ...
  - Live export of most common download locations
    - To be imported into Proxies, Firewalls, IDS
  - Company-wide hash-scan with Encase
- Better virus scanner comparison
  - Automatic re-scan of malware files with each signature update (partly implemented)

Developments and future steps (2)

- Automated analysis:
  - Enhance wine sandbox results
  - rewriting DLLs to log even more
  - Also very interesting:
    - Diploma Project about automated behavior analysis
    - http://pi1.informatik.uni-mannheim.de/diplomas/show/59
    - Extensive API-hooking approach
- MalwareDB
  - A research database for preserving malicious computer programs

Introduction to Malware DB

- ”Fred, where is the DVD with the malware collected in January?”
- ”Somewhere on my desk? ... I was sure that it was laying on my desk...”
- ”I really need that to test something...”

- MalwareDB Scope
  - Simple storage mechanism to archive malware
  - Easy way to tag and classify the malware
  - Multiple interfaces to query and get the malware
  - Not a signature database
  - Not exhaustive

MalwareDB data store (v1)

- MalwareDB only contains metadata for each malware
- Files are stored on the filesystem
- Malware is identified by SHA-2 (256bits)
- For managing collisions (if any), MalwareDB keeps track of:
  - the original filename
  - information about file (like magic code, mime/type...)
- Source is a unique field to identify the origin of the malware
  - who or what is submitting the malware
- MalwareDB supports free tagging for classification, excluded are some reserved prefixes like RFC, CVE, OSVDB,..

Query the MalwareDB

- Using the web interface : http://www.csrrt.org/maldb/index.pl
- Using the RSS feed : http://www.csrrt.org/ml/rss/latest.xml
- Using the DNS interface to check the existence of a malware from its fingerprint:
  - dig -t TXT 3d5a9097cda0565ccc4a0e8aaa703b8543.187 \ 31eb80bce12e8d9958f115fa468.sha1.maldb.csrrt.org
  - 63 bytes have to be separated by a dot to split into “subdomains”, server reassembles accordingly
  - You could use the DNS interface as an RBL-like interface for early detection/warning but don’t forget that the database is not exhaustive.

Conclusion about the MalwareDB

- First try for a malware database (far from being perfect)
- Legal implication (copyright, computer security,...)
- Could be used by attackers as a repository
  (measure must be taken to avoid that)

Conclusions

- Nepenthes provides a nice way to collect malware
- It can also be used to block intruders/malicious URLs
- Early reaction is possible for the attacking vectors implemented in nepenthes
- Signature based systems definitely not fulfilling requirements
- Signature based plus behavioral analysis is definitely a way to pursue
- Automated analysis is a need, especially when receiving large feeds
- Hopefully increased joint-effort for sandbox-alike tools in the future

Thanks to

- mwcollect.org
  - Thorsten Holz, Markus Kötter
  - Paul Baecher, Georg Wicherski
- CSRRT-LU
  - Alexandre Dulaunoy
  - Gerard Wagener
- Hispasec Sistemas (VirusTotal)
  - Julio Canto
- Telecom Italia (Early Warning Team)
  - Gaetano Zappulla

SES ASTRA
An SES GLOBAL Company

Zen and the art of collecting and analyzing malware

Computer Security Research & Response Team

CSRRT-LU

Questions?

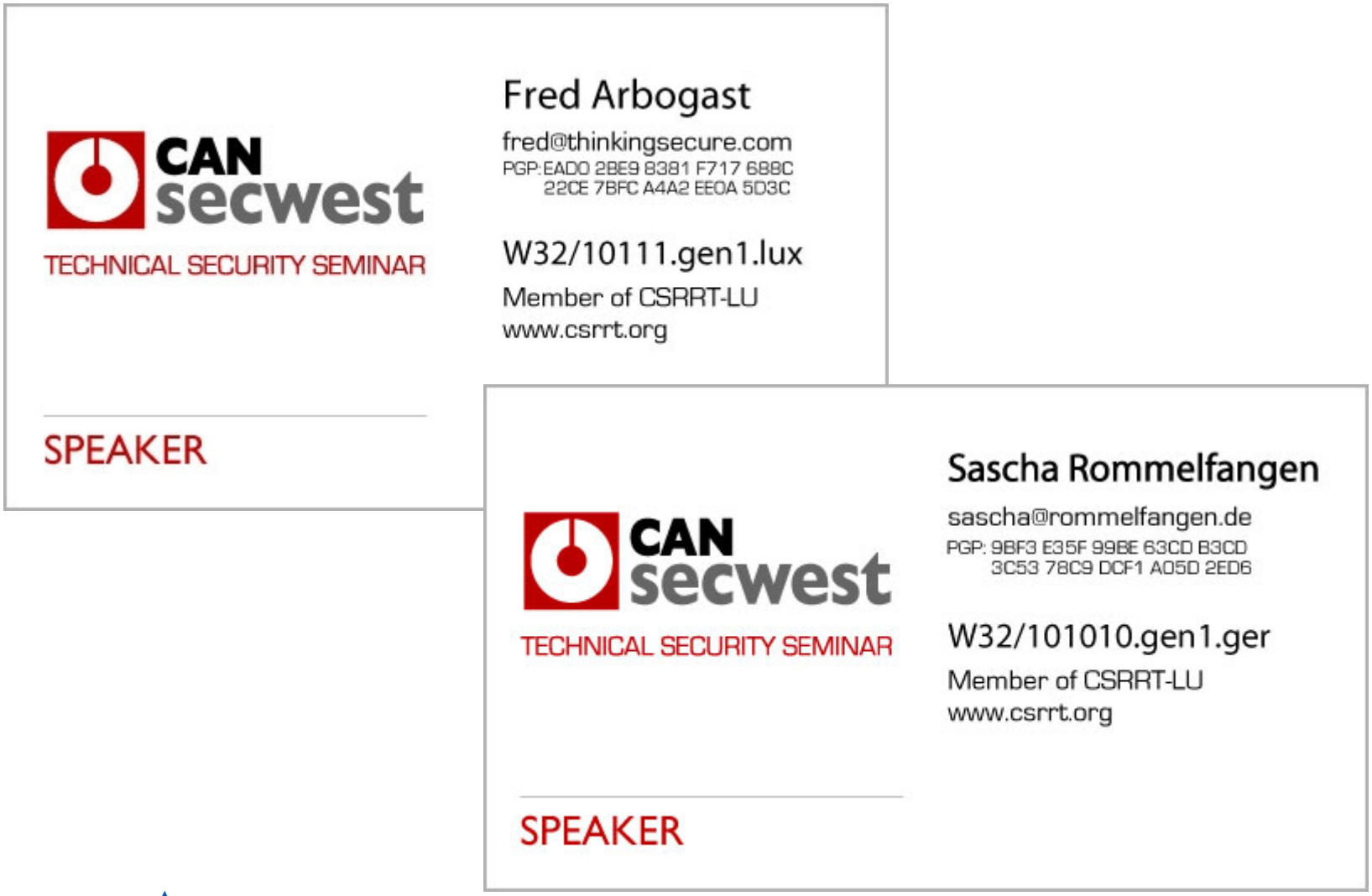
SES ASTRA
An SES GLOBAL Company

Zen and the art of collecting and analyzing malware

Computer Security Research & Response Team

CSRRT-LU

Thank you



SES ASTRA
An SES GLOBAL Company

Zen and the art of collecting and analyzing malware

Computer Security Research & Response Team

CSRRT-LU