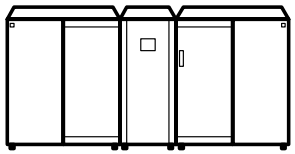
PROTOS

Results

Vulnerability handling

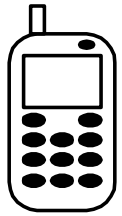
Conclusions

[Server & Telephony]

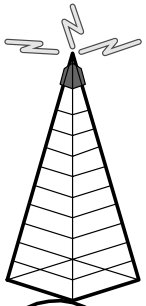


WAP Gateway

[Embedded & Telephony]

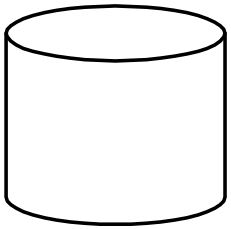


WAP Terminal



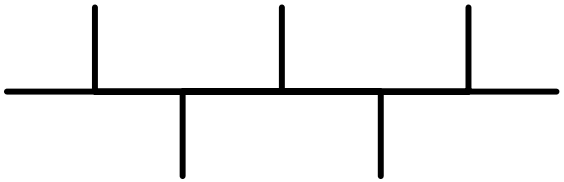
Networks

[Server & Infrastructure]



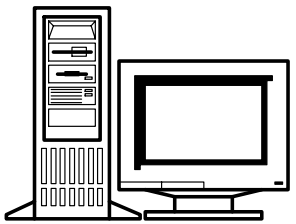
LDAP Database

[Infrastucture & Management]



SNMP

[Home & Desktop]



HTTP Client

Test-suite summary

Test-suite	Test groups	Test cases	Failed products
wap-wsp-request	39	4236	7 (7 tested)
wap-wmlc	84	1033	10 (10 tested)
http-reply	115	3966	5 (12 tested)
ldapv3	93	12649	6 (8 tested)
snmpv1	118 / 100	29516 / 24100	12 (12 tested)

PROTOS Cycles

- 1999: Cycles 1-3, test methodology prototyping, from brute-force to informed test design
- 2000: Cycles 4-5, from IETF/BNF towards telecom/ASN.1, proof of concept test-suites released
- 2001: Cycles 6-7, capability to handle telecom complexity, more support for complex protocols and other descriptions besides BNF

A case study - PROTOS/c04-wap-wsp-request

- We applied the proposed model in practice:
  - Wireless Application Protocol (WAP) Suite
    - The mobile counterpart of WWW
  - A test-suite was developed:
    - For the WAP Wireless Session Protocol (WSP) akin to HTTP
    - More specifically for the WSP-requests akin to HTTP-requests
    - WSP-requests are processed by WAP-gateways akin to HTTP proxies and servers
    - 4236 test-cases were generated through syntax testing:
    - Seven easily available WAP gateway products were tested

[http://www.ee.oulu.fi/research/ouspg/protos/testing/c04/wap-wsp-request/]

Test-suite results

- Results for the 7 WAP gateway products:

Testrun #	Total test-cases	Failed test-cases	Total groups	Failed groups
tr-001	4236	569	39	10
tr-002	4236	141	39	18
tr-003	4236	10	39	2
tr-004	4236	385	39	16
tr-005	4236	664	39	8
tr-006	4236	622	39	14
tr-007	4236	148	39	20

- 7 out of 7 implementations failed
  - Some implementations failed in 50% of anomaly groups
    - 50% chance of monkeys creating a Shakespeare sonnet!
  - Failures (read vulnerabilities) were reported to the vendors

Recent PROTOS Test-Suite: c06-snmpv1

- **CERT® Advisory CA-2002-03 Multiple Vulnerabilities in Many Implementations of the Simple Network Management Protocol (SNMP)**
  - http://www.cert.org/advisories/CA-2002-03.html
  - Couple of man months to develop
  - Several man months to coordinate
  - As of May 2002:
    - **over 200 vendors informed**
    - **~140 vendors have responded publicly**
    - **~100 vendors had affected (vulnerable) products**
  - New vendor statements keep pouring into the advisory

Test-suite vs. bug hunters

- *"… Imagine if FTP was assumed to be free of exploits and somebody dumped a tool on the Internet that demonstrated all the discovered vulnerabilities all at once."* [snmpv1]
- *"X will deliver the patch (fixed 19 bugs) ..."* [ldapv3]

- The test-suites do not discover just one, but a set of vulnerabilities in the interface

Surprising findings in sub-components

- *"Very interesting. We were extremely careful, but there was a deeply embedded support routine that was not doing proper bounds checking on the host portion of the URL." [wap-wsp-request]*

- Even with careful software development, portions that were outside the process can contain failures
- Also software implemented in Java were shown to have buffer overflows in the native code sections

Surprising return packets

- *"The most serious problem (from a security point of view) might cause the gateway to transmit some of its memory contents as an HTTP header name to the HTTP server, though you may not have noticed it doing that." [wap-wsp-request]*

- Sometimes the software does not fail in noticeable form, but just returns some (confidential?) data or even memory structures to the requester

Test reproduction

- *"It is always good to receive reports on the performance of our products, especially when they provide details on how to reproduce problems." [wap-wsp-request]*

- Test-suites provide the vendor the means for assessing the quality of the product themselves

Regression testing

- *"I believe this alert will do wonders for improving general security in LDAP implementations."* [ldapv3]

- If integrated to the software development process, the test-suites have a chance of ‘raising the bar’ in the software products
- The most trivial errors are easily discovered and eliminated

Code reuse: bugs are in the hiding

- *"[...] I loaded the oldest backup tape I could find and read, which was from early 1991, and some of these vulnerabilities were present then […] these vulnerabilities have been silently present for over a decade and they are ubiquitous […]"* [snmpv1]

- A bug in software can be in the hiding, and be copied into new instances and versions of the software

Motivation for quality improvement

- *"I am disappointed in X for not even testing for these vulnerabilities until pressure was put on them through resellers and for not publicly announcing it so that administrators are made aware."* [ldapv3]

- Public pressure to reliability and security issues increases

Product comparisons

- *"I was wondering if you are going to post your results anywhere for us to look through? We would be interested to see how we compared to the other products you have been testing." [wap-wsp-request]*

- Both the vendors themselves and the customers lack the means of comparing product quality between products of different vendors

General observations

- Death-Zones were apparent
  - Several products had problems in exactly same categories
    - E.g. String table index handling in wap-wmlc and proxy authentication in http-reply were rather error prone
- Decoder problems are abundant
- Real diversity is not produced by developing different implementation with same toolkits, but by using different tools and paradigms
- One ancient open source implementation will affect several implementations of the future
  -> Bugs will persist due to the code reuse

Vulnerability handling: Communication

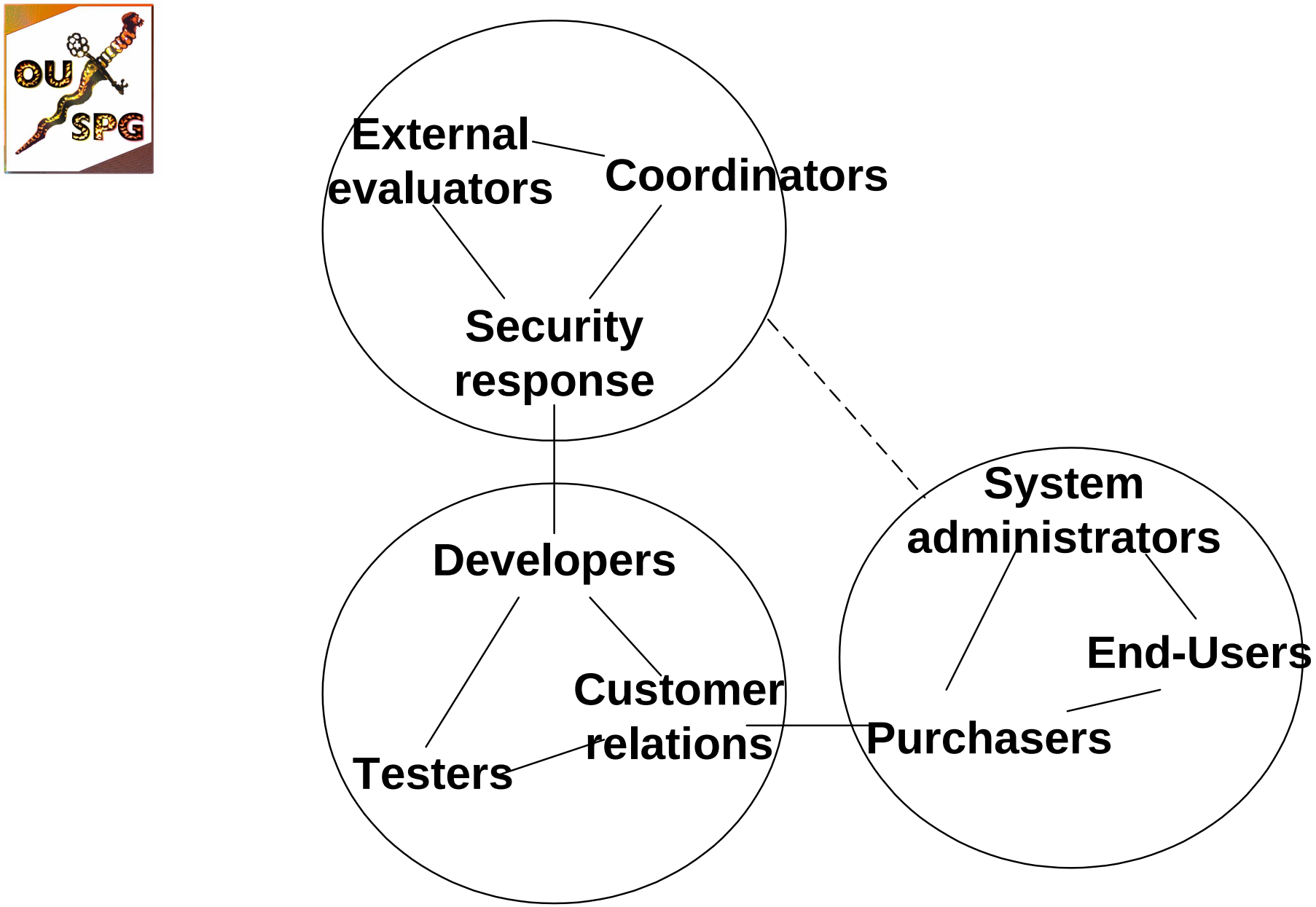
Test-suite	Failed products	Vendor responses	Advisory
wap-wsp-request	7 (7 tested)	5	n/a
wap-wmlc	10 (10 tested)	1	n/a
http-reply	5 (12 tested)	2	n/a
ldapv3	6 (8 tested)	10	CA-2001-18
snmpv1	12 (12 tested)	~140	CA-2002-03

Bug reporting

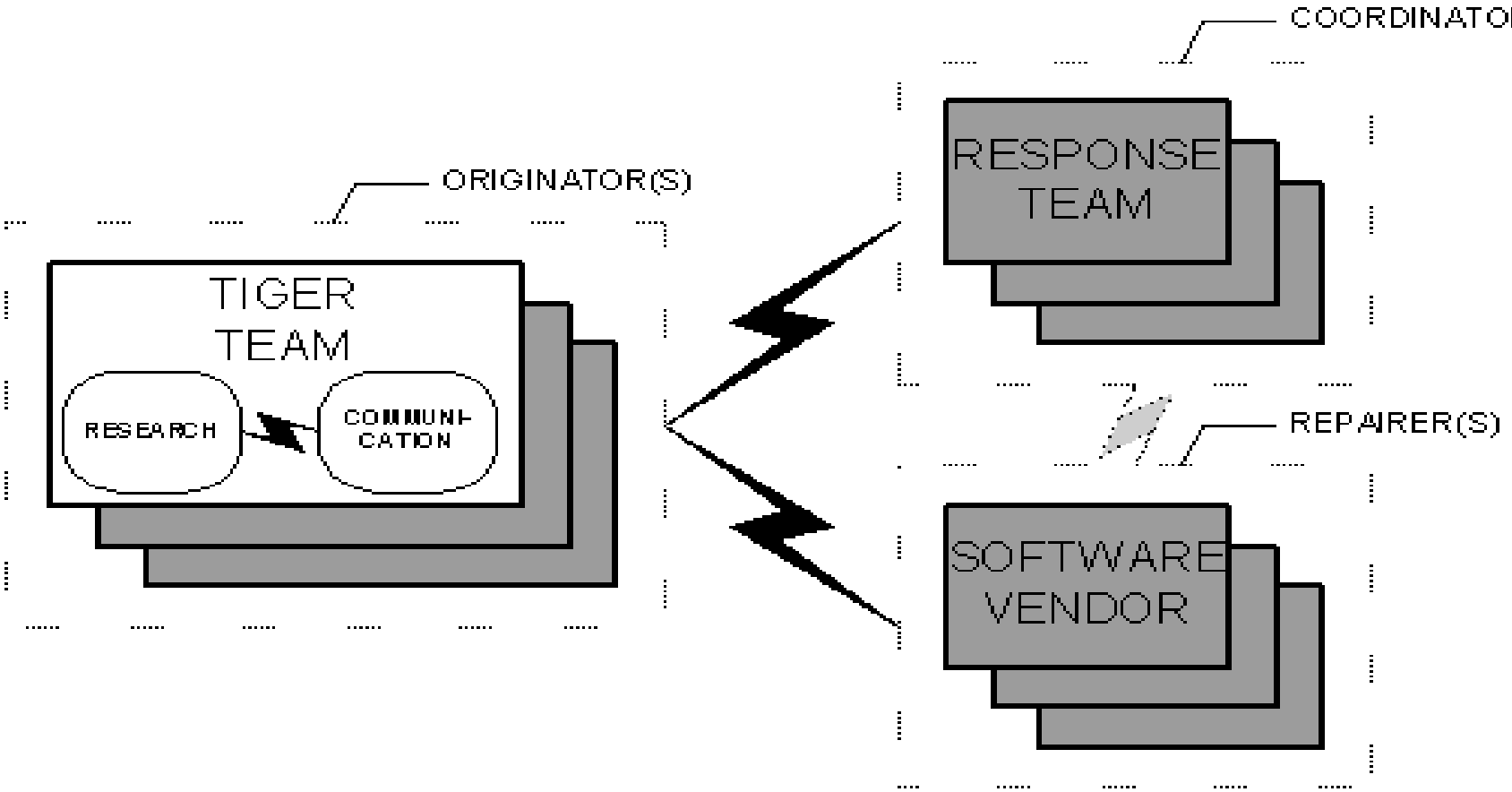
- In order to resolve vulnerabilities, we have been in touch with:
  - Sun Microsystems, SGI, IBM, HP, Legato, Insignia, Microsoft, Netscape, Corporate Time, Diva, Audicode, Infinite, Nokia, Integra Microsystems, Peramon, Kannel, Silicon AS, Ericsson, Ausys, Phone.com (OpenWave), Benefon, Siemens, Lotus, NAI, Oracle, iPlanet, TeamWare (FujitsuSiemens), OpenLDAP, Mozilla, Opera Software, 3Com, Lantronix, Novell, SNMP Research International, and Computer Associates

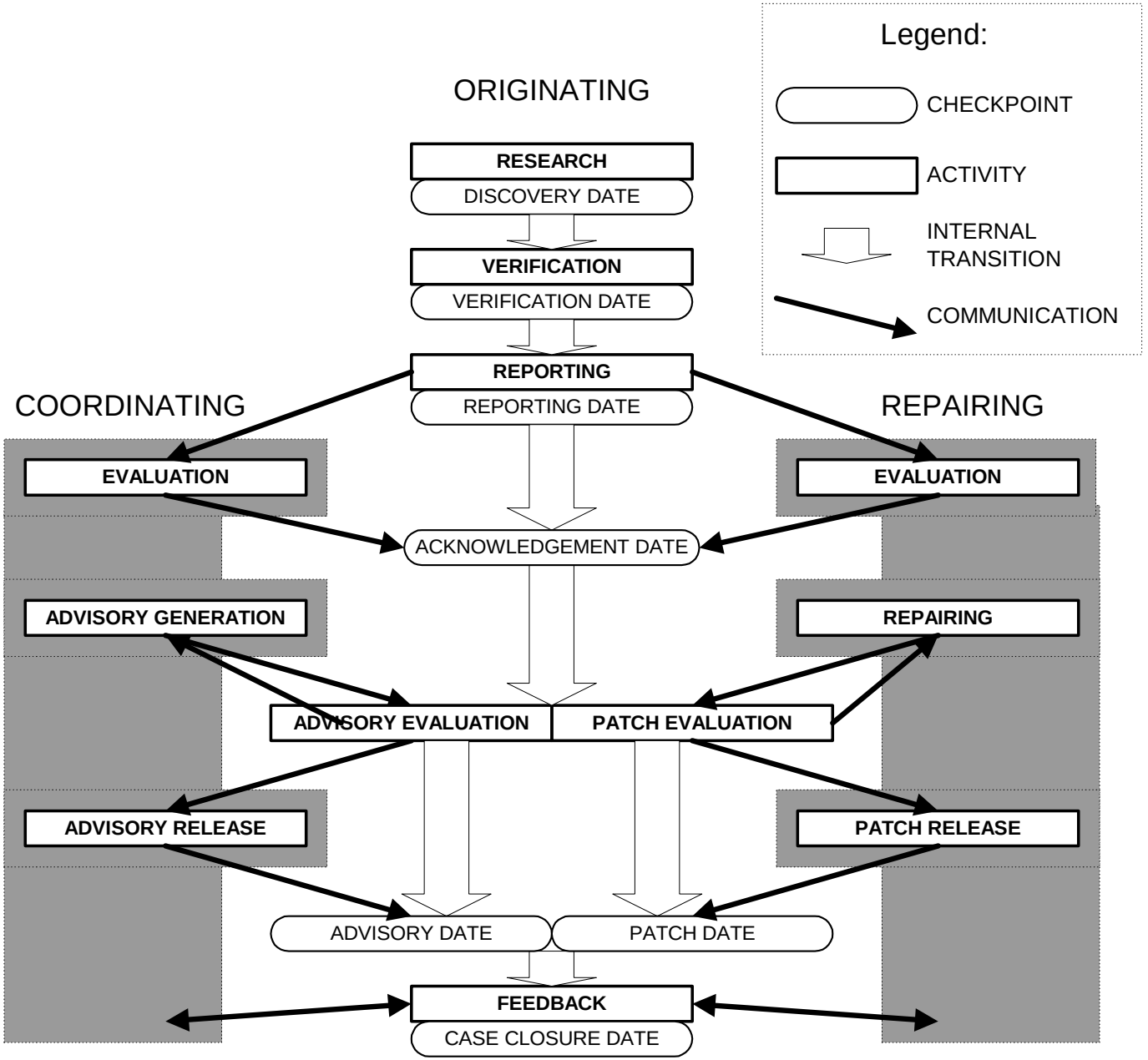
The Problems - as we see them

- Increasing complexity and poor quality in software:
  - The sheer number of information security vulnerabilities
  - Caused mainly by well-known trivial programming errors
  - End-result is a vicious patch-and-penetrate cycle
- Inefficiencies of traditional vulnerability process:
  - Volume of communication, reproduction problems
  - A slight variant of the same exploit may bite multiple vendors
  - Reappearance of same bugs, regression testing
  - Caveat emptor - customers should be able to evaluate
- No-disclosure vs. Full disclosure debate:
  - Effort might be better spent on resolving the actual issues


External evaluators
Coordinators
Security response
Developers
Testers
Customer relations
System administrators
End-Users
Purchasers

Three roles in the vulnerability process




Legend:
CHECKPOINT
ACTIVITY
INTERNAL TRANSITION
COMMUNICATION
ORIGINATING
RESEARCH
DISCOVERY DATE
VERIFICATION
VERIFICATION DATE
REPORTING
REPORTING DATE
COORDINATING
REPAIRING
EVALUATION
EVALUATION
ACKNOWLEDGEMENT DATE
ADVISORY GENERATION
REPAIRING
ADVISORY EVALUATION
PATCH EVALUATION
ADVISORY RELEASE
PATCH RELEASE
ADVISORY DATE
PATCH DATE
FEEDBACK
CASE CLOSURE DATE

Conclusions = Reality Check

- Single vulnerabilities are not ‘significant’:
  - Bugs come and go
  - Small human error (e.g. snprintf() -> sprintf), huge impact
- Underlying reasons:
  - Growing complexity
  - Focus on higher level issues (crypto, PKI ... -> trust)
- What are we worried about?
  - Patch & Penetrate -> Media numbness
  - Public ‘trust & confidence’ in new technologies
  - Embedded world:
    - Lack of diversity
    - Patch deployment probs

As easy as making the pigs fly?

- Real Life™ processes are complex
- Professional approach required and promoted

Questions and Feedback:

http://www.ee.oulu.fi/research/ouspg

ouspg@ee.oulu.fi