ERNW
Living Security.

Vulnerability Assessments

Enno Rey
erey@ernw.de

Security Assessments

- **There's different types of them (you already knew that, didn't you? ;-)**

- **The differentiating factor might be**
  - Scope
  - Tools
  - Approach/Methodology
  - Format/structure of the report

- **The *main* differentiating factor is what we call the**
  - "Scientific objective" (the assessment's objective)
  - Think: the question to be answered by performing the assessment.

Classification as of ERNW's services

- **Penetration Tests (Pentests)**
  - With the means of an attacker, within a given period of time
  - Might discover product or implementation flaws, but not "big picture".
- **Audits**
  - Check if state-as-is is compliant with some "defined state"
  - Typically looking "at the systems" (e.g. config. details) + documentation
- **Reviews**
  - Usually based on documentation (+ some implementation audit)
  - Often before deployment ("do you think we can run it this way?")
- **Evaluations**
  - Mostly focused on products (=> our client might be vendor)
  - Purest form of "assessing an item"

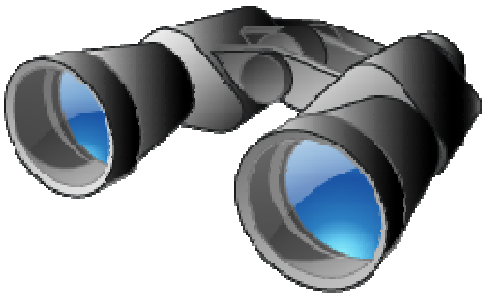
Scientific Objective/"The question", Examples

ERNW Living Security.

- **Can a skilled and motivated attacker break into our network?**
- **→ Pentest**

- **Which vulnerabilities does this system have?**
- **→ Vuln. Assessment**

- **Are our configuration guidelines followed?**
- **→ Audit**



Scientific Objective/"The question", Examples

ERNW Living Security.

- **In this project, are we on the right track as for security?**
- **→ Review**

- **Does this product dispose of a given set of specified security properties?**
- **→ Evaluation**

Vulnerability Assessments

- **… do exactly that: they assess vulnerabilities ;-)**
- **→ At first you have to define (or understand) what a vulnerability is. More on this on next slide.**
- **A pentest can be part of a vulnerability assessment. But (usually) not the other way around.**
- **There's no standard procedure for vulnerability assessments.**
  - But certainly some "commonly performed steps".

Threats & Vulnerabilities

ERNW Living Security.

Threat	Vulnerability
Trap	Desire for cheese and a wimpy neck
Theft	Open door and no security guard
Information Disclosure	Clear-text transport in public networks
Unauthorized access	Weak authentication

Threat:

"A threat has the potential to harm assets such as information, processes and systems and therefore organizations."
(ISO 27005)

Vulnerability:

A "vulnerability does not cause harm in itself, as there needs to be a threat present to exploit it."

Do vulnerability assessments make sense?

ERNW Living Security.

- **For enterprises? Not in the strict sense.**
  - It's all about risk anyway.
  - Vulnerability = not risk!
  - Still, term (and procedure) used in enterprise space.
    - Remember: common use of these terms is wishy-washy anyway.

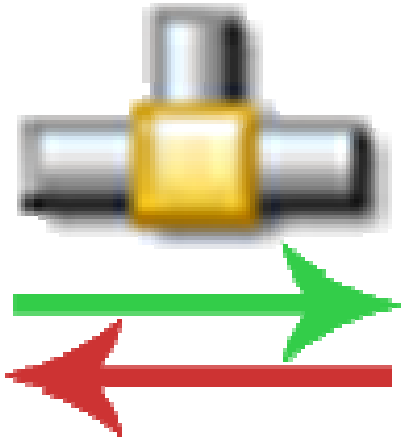
- **For hardware/software vendors? Yes, sure!**
  - From the three risk composing factors it's (mostly) the only one they can work on.

- **So let's talk about *how* to do this stuff potentially…**

Some disgression on “vulnerability”.

ERNW Living Security.

- **First, agree on definition what a “vulnerability” is, in the given context**
  - An open port?
  - An open port with an unauthenticated protocol?
  - An open port with an authenticated protocol, but a “weak” default authenticator?



On the role of threats (when performing vulnerability analysis)

ERNW Living Security.

- **Thinking about threats might be helpful.**
- **What's the threat(s) the vulnerability will help to materialize?**

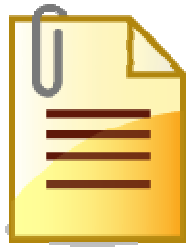
  - Attacker? Operator error?
  - Unfortunately, reducing the vulnerability for the first might increase the vulnerability for the second. Note how important the threat angle is?
  - I'll introduce my *personal ceterum censeo* of this workshop for the first time here. Regardless of the just cited apparent prioritization conflict:

    Shipping devices with (community-based) SNMP enabled and community strings *public* and/or *private* is ALWAYS a bad idea!

Concept of “relevant vulnerabilities”

ERNW Living Security.

- **Brings in the “risk based point of view”.**
  - In most cases will only be doable for “typical deployment scenarios”.
  - Do you know those (reported back from the field)?
  - Sorry for asking this. Of course you fully understand how your customers are using your stuff, being a good customer-oriented company ;-)



- **Identify those vulnerabilities that could lead to high risks.**
  - Structured approach of assessing risks needed, see later session.
  - Might have the additional benefit of being able to document mitigating controls on the configuration/operations level. For your customers ;-)

Back to/more on vulnerability assessments

- **So now, after having defined your set of vulnerabilities to be assessed, how to do that.**
- **In the following a first attempt to discuss**

Approaches

- **On the code level.**
- **On the configuration level.**
- **On the behavior level.**
- **By “actively targeting a system”.**
- **A note on “backdoors”.**

Approaches, Code level

- **Perform security assessment on the code level**
  - Code reviews
    - Expensive
    - Might be required anyway if striving for CC evaluation of a certain level.
  - “Statistical approaches” (lines of code, complexity of code as of certain metrics [McCabe et.al]).
    - Usually much easier to perform than partial/full code review.
    - Less value though, in particular when it comes to mitigation.

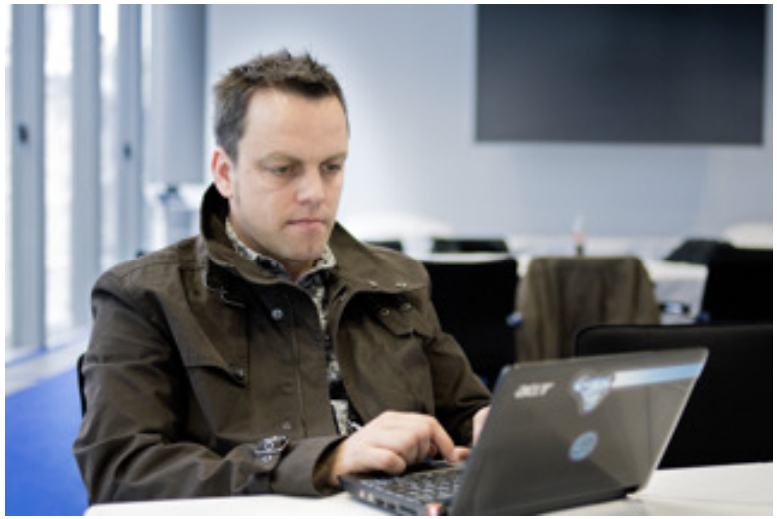
Configuration level

- **Have your homework done (definition what constitutes a vulnerability, see above).**
  - Btw: yes, default enabled SNMP with *public/private* IS a vulnerability ;-)
- **Then perform audit (*as-should* vs. *as-is*).**
- **Can (obviously) only be performed for *default* state.**
  - Supply chain might change picture.
  - Operational/ “setup” practice in your customers’ environments will (again: obviously) play a large role later.
    - Disabling SNMP not too helpful if all of them immediately re-enable it anyway. (and might even lose you customers as this annoys them. Think Win Vista UAC.)
    - → Understand their operational needs and practices. Provide guidance where necessary.
- **Should be easy to define “intelligent KPIs” in this space.**

A note on the supply-chain aspect

- **Becomes more and more important for device security in critical infrastructures.**

- **See upcoming presentation of Graeme Neilson and Enno Rey at Day-Con IV for more on this.**



On the behavior level

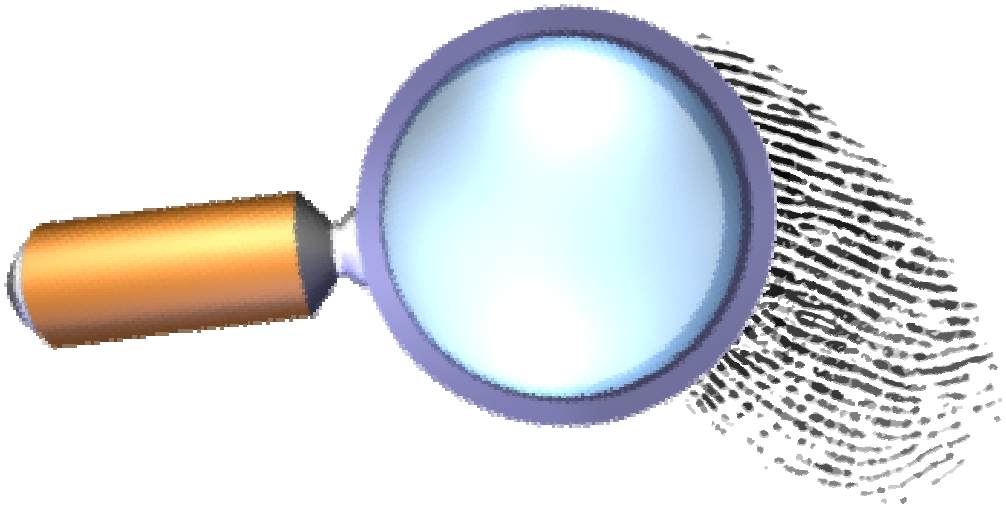
- **Monitor network behavior of system(s) in question**
  - By inspecting packets' contents.
  - ⇔you should expect current backdoors to use encrypted communication anyway.
  - Still, statistical methods might kick in.

  - Only makes sense for organizations deploying your stuff.

By “actively targeting a system”

ERNW Living Security.

- **Vulnerability scanning**
  - Can be automated → easy.
  - Unfortunately, for your kind of systems, will only discover stuff you should have (hopefully) found/avoided earlier on anyway.



- **Fuzzing**
  - We highly recommend that.
  - Might require creation/adoption of tools. Usually it’s worth it.

How to rate security of closed source SW

ERNW Living Security.

- **Some reference to Michael (Thumann's) talk.**

Backdoors

ERNW Living Security.

- **[PAXSON00]:**

- **“A backdoor is a mechanism surreptitiously introduced into a computer system to facilitate unauthorized access to the system.”**

Backdoor, Definition II

- **This brings up some interesting questions.**

- **Is enabled SNMP with public/private a backdoor?**
  - Based on deliberate decision (= “surreptitiously”?).
  - Means to provide access.
  - Well, yes, maybe not intended for *unauthorized* access.
  - But used for such in many cases…

Types of backdoors

- **Buffer Overflow vulnerability**
- **Hidden configuration options (“allowHiddenAccountLogin=YES”)**
- **Unsecure cryptographic properties**
  - Weak initialization vectors
  - Manipulated S-Boxes (e.g. in AES)
  - Deterministic PRNG
- **Master password (“lkwpeter”)**
- **Hidden credentials (user/pass)**
- **Port knocking**
- **Data leakage/logging second channel (external system, …)**
- **Additional access mechanism (SSH, telnet, …)** ← ***typical rootkit behavior***