DFN CERT

NoAH Honeynet Project

NoAH

European Network of Affined Honeypots

17th TF-CSIRT Event
23/24 January 2006
DFN-CERT Services GmbH

Introduction

- NoAH is a Specific Support Action in the Sixth Framework Programme of the European Union.
- Start: April 2005
- End: 31 March 2008
- Homepage: http://www.fp6-noah.org/
    - 1st NoAH Workshop: May 2006

Introduction

- Project partners
  - Foundation for Research and Technology Hellas (FORTH) - Coordinator
  - Alcatel CIT
  - DFN-CERT Services GmbH
  - Eidgenössische Technische Hochschule Zürich (ETHZ)
  - Hellenic Telecommunication and Telematics Application Company S.A (FORTHnet)
  - Trans-European Research and Education Networking Association (TERENA)
  - Virtual Trip Limited
  - Vrije Universiteit Amsterdam (VU)

Introduction

- Main objectives
  - Design a distributed state-of-the-art infrastructure of honeypots.
  - Develop techniques for the automatic identification of attacks, and for the automatic generation of their signatures.
  - Installation and operation of a pilot honeypot infrastructure.
  - Distribution of open-source software, anonymised attack data and signatures to NRENs, ISPs, and CSIRTs.

Work Packages

Finished Work Packages:

- WP0: Requirements Analysis and State-of-the-Art
  - WP0.1: Review existing technology.
  - WP0.2: Identification of the requirements of the NoAH infrastructure.
  - Deliverables D0.1 and D0.2 available on NoAH's websever

Work Packages

Running Work Packages:

- WP 1: Design of System Architecture
  - Specification of NoAH's honeypot components, the infrastructure, and signature generation mechanism.

Comming Work Packages:

- WP2: Implementation
  - Implementation of the NoAH's honeypot components and infrastructure
- WP3: Demonstration and Pilot Operation
  - Operation of the pilot infrastructure in conjunction with a number of participating sites.

Preliminary Results

Architecture Requirements:

- Detection of zero-day attacks and worms
  - Avoiding false-positive results.
  - Detection has to be reliable.
  - Detection of worms in an early stage of spreading.
- Well-suited to capture data for automatic signature generation.
- Scalability
  - Efficient cooperation with NRNs, CSIRTs, and ISPs.
  - Easy and secure deployment of NoAH components.

Preliminary Results

Resulting Solution: Hybrid architecture composed of low- as well as high-interaction honeypots

- Motivation: Combination of advantages of both types of honeypots to fit all requirements.
  - High accuracy of attack detection (HI honeypot)
  - High potential to capture data (HI honeypot)
  - High scalability of architecture (LI honeypot)

Preliminary Results

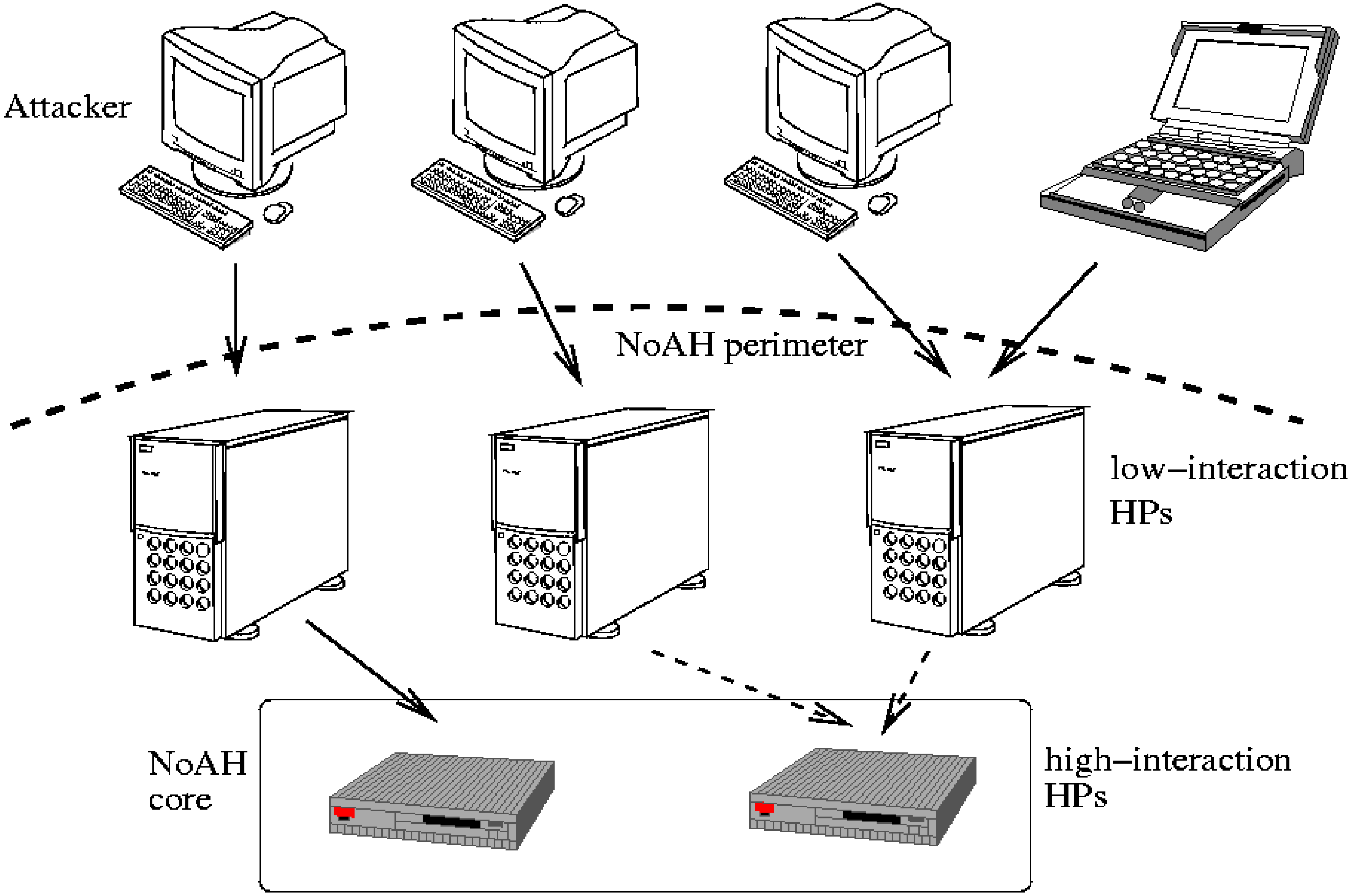
DFN CERT

Recapitulation: Architecture Requirements:

- Detection of zero-day attacks and worms (→ HI honeypot)
  - Avoiding false-positive results.
  - Detection has to be reliable.
  - Detection of worms in an early stage of spreading.
- Well-suited to capture data for automatic signature generation (→ HI honeypot).
- Scalability (→ LI honeypot)
  - Efficient cooperation with NRNs, CSIRTs, and ISPs.
  - Easy and secure deployment of NoAH components.

Preliminary Results

NoAH Architecture



Preliminary Results

- Low-interaction honeypots (e.g. honeyd)
  - Accept connections from attackers.
  - Proxy connections to high-interaction honeypots.
  - Performance to cover broad IP space to increase detection probalility of zero-day attacks and worms.
  - Easy and secure deployment by participating sites (much better acceptance compared to high-interaction honeypots).
  - Potential for filtering out known attacks.

Preliminary Results

- High-interaction honeypots:
  - Providing differnt services (e.g. HTTP server)
  - Deployment of „Argos“ containment environment (Vrije Universiteit Amsterdam)
    - Detect attacks that inject data to modify execution control flow (EIP register) – e.g. almost all exploits for buffer overflow, format string, and double-free vulnerabilities.
      - Dynamically taint all network input (e.g. HTTP-Requests).
      - Prevent and detect if tainted data is used in an illegitimate way – e.g. used as function pointer or load into EIP register.
    - Attack is stopped before it can get in control of the honeypot.
    - Potential of tracking attack related memory flows.
    - Cope with polymorphic shellcode.
    - Capture of exploit integrated shellcode.
    - Capture of attack related data.

Preliminary Results

Signature generation

- Based on data from high-interaction honeypots (e.g. Argos) and network traffic (host and network based).
- Detection of polymorphic attacks
- Introduction of Meta Signatures
  - Composed of multiple types of signatures.
  - Includes flag to indicate polymorphism.
  - Motivation: Combination of different types of signatures are better suited to detect polymorhic attacks.

Thank You

? ? ?