Industrial Control Systems Honeypot

May1601

Aashwatth Agarwal
Dan Borgerding
Jon Hope
Nik Kinkel
Jon Osborne
Korbin Stich

http://may1601.sd.ece.iastate.edu

Client: Alliant Energy
Advisor: Dr. Doug Jacobson

April 28, 2016

Threat Overview

- Highly critical threat
- Advanced attackers
- First attack on a power grid (Ukraine)

Project Overview

What is a honeypot?

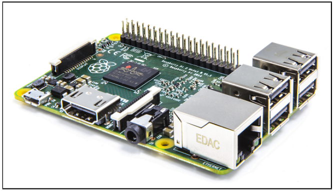
A security mechanism designed to detect, deflect or counteract attempts at unauthorized use of information systems.

Purpose

- Trick intruders
- Alert administrators
- Detect attack vectors
- Prevent data loss/corruption

The Deliverable

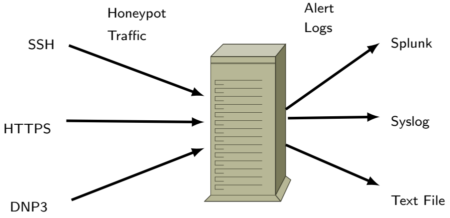
- Customized honeypots for multiple protocols
- Minimal IDS
- Automated deployment & management
- Configurable logging backends
- Cheap, plug & play device



Raspberry Pi 2

Tech Challenge 1: Dealing with Lots of Protocols

- Many honeypot protocols and logging backends to deal with
- New protocols must be integrated quickly and safely



Design 1: Honeypot Plugin Framework

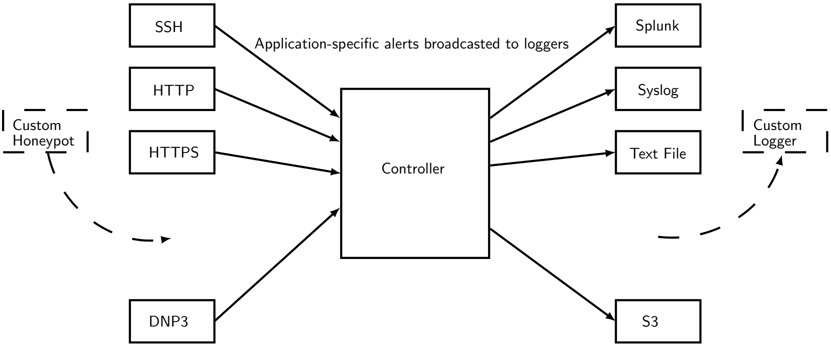


Figure: Multi-process, message-passing architecture

pluggable · concurrent · separate address space · easy testing

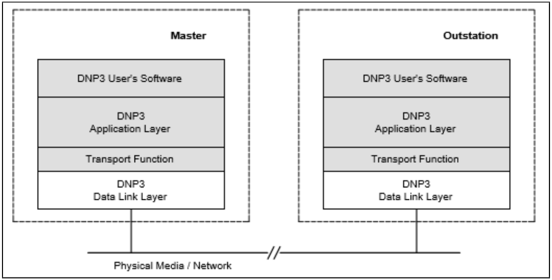
Demo: Honeypot Plugin Framework

Demo

Tech Challenge 2: Obscure SCADA Protocols

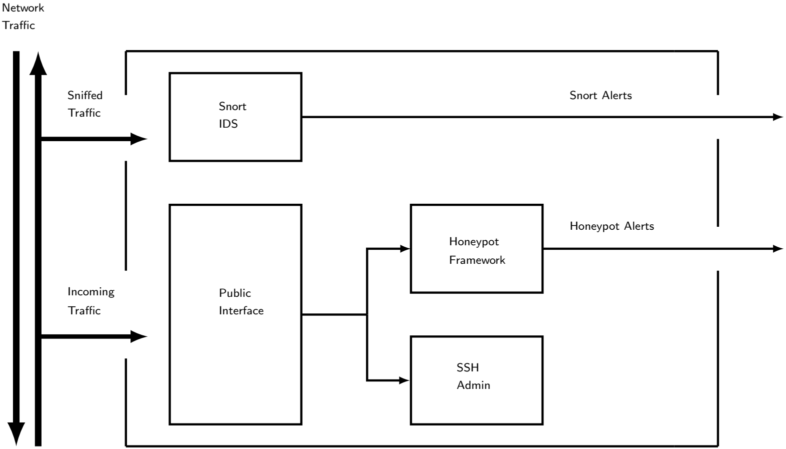
DNP3

- Application layer protocol built on TCP/IP
- Consists of Data, Transport, and Application layers
- Testing
- Secure Authentication



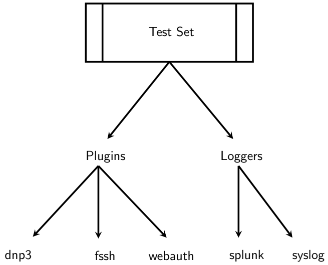
DNP3Spec-V1-Introduction-20071215

Design 2: Device Architecture



Simplified Device Internals

Testing 1: Unit Tests

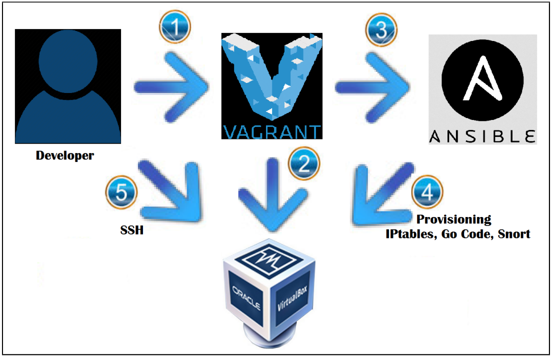


- Unit Testing
- Code Output Verification
- Plugin Strategies
- Log Strategies
- Core Strategies

Testing 2: Integration Testing

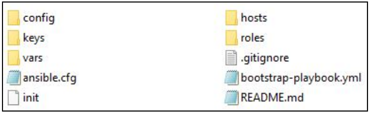
Vagrant

- Repeatable environment simulation
- Automatic streamlined VM Provisioning



Vagrant Environment

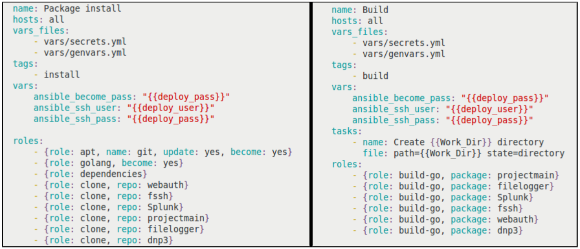
Tech Challege 3: Simultaneous, Multi-Site Deployment



config
keys
vars
ansible.cfg
init
hosts
roles
.gitignore
bootstrap-playbook.yml
README.md

Deployment Directory

- 28 Devices.
- Numerous Locations
- **Ansible Makes This EASY**



```
name: Package install
hosts: all
vars_files:
    - vars/secrets.yml
    - vars/genvars.yml
tags:
    - install
vars:
    ansible_become_pass: "{{deploy_pass}}"
    ansible_ssh_user: "{{deploy_user}}"
    ansible_ssh_pass: "{{deploy_pass}}"

roles:
    - {role: apt, name: git, update: yes, become: yes}
    - {role: golang, become: yes}
    - {role: dependencies}
    - {role: clone, repo: webauth}
    - {role: clone, repo: fssh}
    - {role: clone, repo: Splunk}
    - {role: clone, repo: projectmain}
    - {role: clone, repo: filelogger}
    - {role: clone, repo: dnp3}
```

```
name: Build
hosts: all
vars_files:
    - vars/secrets.yml
    - vars/genvars.yml
tags:
    - build
vars:
    ansible_become_pass: "{{deploy_pass}}"
    ansible_ssh_user: "{{deploy_user}}"
    ansible_ssh_pass: "{{deploy_pass}}"
tasks:
    - name: Create {{Work_Dir}} directory
      file: path={{Work_Dir}} state=directory
roles:
    - {role: build-go, package: projectmain}
    - {role: build-go, package: filelogger}
    - {role: build-go, package: Splunk}
    - {role: build-go, package: fssh}
    - {role: build-go, package: webauth}
    - {role: build-go, package: dnp3}
```

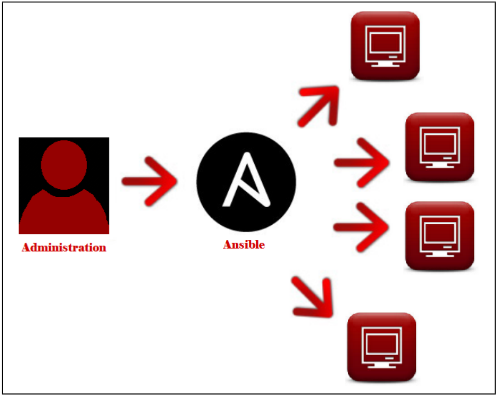
Ansible Honeypot Administration

Demo: Provisioning with Ansible

Demo

Long-term Support, Administration, and Maintenance

- Update process must be:
  - flexible
  - single-step
  - fault-tolerant
  - idempotent
- Manual administration option necessary
- Auto-notify for security updates



Ansible Updates

Questions