***Investigating BGP security* Top Level Design**

|  |  |
| --- | --- |
| Document Owners | *Roey Maor and Michal Cohen* |
| Instructor / Guide | *I**tzik Ashkenazi* |
| Table of contents | Introduction – 2  Project general architecture – 3  Management overview – 4  Project planning – 5  Appendix - 6 |
| Date | 04.12.16 |
| Document Version | 1.0 |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

# Introduction

## 

## Background

The internet is composed of tens of thousands of Autonomous systems (AS). An AS is a large group of connected machines, usually representing an organization or a geographic area. Each AS decides on its intra-routing policy on its own independently from other ASs. These routing protocols aren’t scalable and differ between the different ASs. Unlike the intra-routing protocols, there is only one de-facto routing protocol between ASs and it is the BGP. The BGP algorithm purpose is to control how packets are to be routed between ASs. It is important not only from a technical aspect, but also from a business point of view.

It was written in the days where security wasn't considered, therefore there are many points of weakness to this protocol. For example, by spoofing fake AS-PATH, one can make packages redirect from their legitimate path through I ts AS.

The BGP algorithm is a path vector algorithm which has no metric function, and is affected by many parameters. Some of the parameters being considered are:

* AS-Path: the ASs identity in the path to a subnet
* The policy of the AS
* The resolution of the subnet prefix
* BGP origin
* And more

## Overview

The main idea of the project is to experiment with the relatively new technology from cisco, which is the VIRL simulation software. To use the latter in order to simulate BGP attacks, study these attacks and reach general conclusions regarding them such as:

* How easy is the execution of the attack from the attackers stand point?
  + In what manner did the ISP needs to cooperate with the attacker?
  + Does the attacker need to deal with race conditions?
  + Does the attacker have to use special hardware?
  + How much code does the attack consume?
* What is the radius of influence (how many ASs can be infected)?
* How long does it take the attack to propagate?
* In which situations such an attack is impossible?
* In which scenarios such an attack is relatively easy?
* How easy is it to detect BGP attack?
* Is it easy to track the origin of the attack (Is it even possible)?
* What measures could AS manager take to prevent such an attack? Make it more unlikely?

The attack we will simulate will be "Black hole" attack, which means the packets are hijacked to the AS of the attacker, but are not re-transmitted to the original target. Such an attack cannot be held for long because many users notice they cannot connect to a certain part of the internet, and therefore investigations will start quickly and the attack will be stopped. A more sophisticated attack is "MITM" attack, in which the packets are retransmitted to the original target. This behavior is much more difficult to achieve because the internet is practically assigned to transmit these packets to the attacker and not to the original target.

An attempt to perform such an attack must make the internet look "asymmetric" in a BGP manner: at least one path to the target must stay legitimate in order to send the packets to the target and not receiving them back. We will try to simulate such an attack on a small topology in VIRL.

## References

* The tutorial of cisco to VIRL: <http://virl-dev-innovate.cisco.com/vsb.php>
* Previous project using CREATE: <https://www.isi.deterlab.net/file.php?file=/share/shared/BGPhijacking>
* Example of a live BGP attack from the DEF CON conference: <https://www.youtube.com/watch?v=oWdjsfsS_Do>
* "Internet networking" webcourse: <http://webcourse.cs.technion.ac.il/236341/Winter2016-2017/>
* The youtube channel of cisco VIRL: https://www.youtube.com/channel/UC41WuzXlJCGY5qLsuZ8aHkQ

## Terminology and Acronyms

|  |  |
| --- | --- |
| Term | Meaning |
| BGP | Border gateway protocol |
| AS | Autonomous system |
| VIRL | Virtual internet routing lab |
| VM-Maestro | The client side of the VIRL software |
| Routem | VIRL control plane traffic generator |
| Ostinato | VIRL data plane traffic generator |

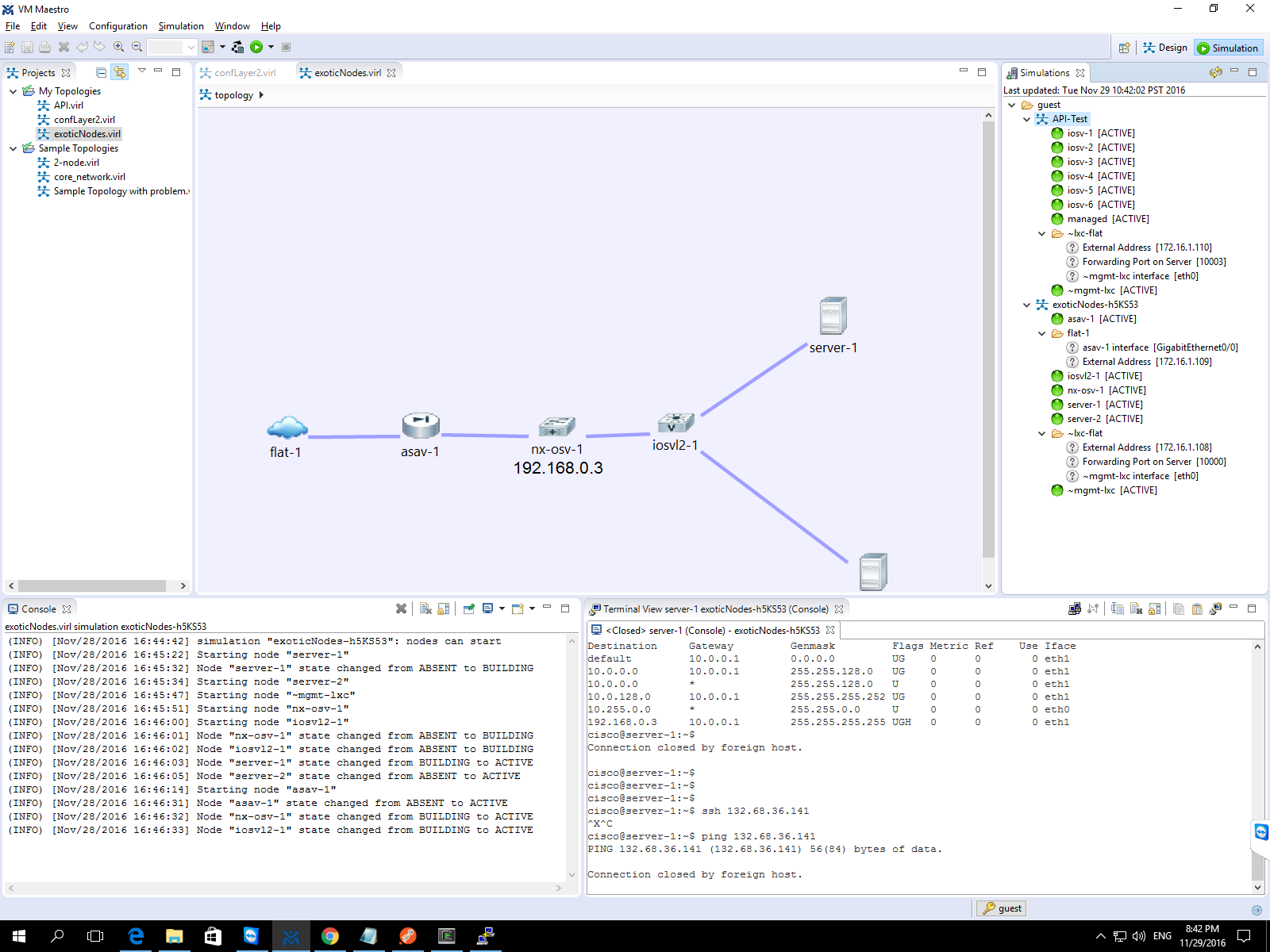
|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

# Project General Architecture

The project will be held in a server in the networking lab at the Technicon. That server runs the VIRL server side software that its main purpose is to execute the simulations that are designed in the VIRL client side (VM-Maestro).



There are three types of ASs:

* Stub AS: an AS connected only to one other AS that provides the first with internet access.
* Multihomed AS: an AS connected as a client to several AS that provides it with internet access. The decision which AS to use in order to rout packets depends on the policy of the AS (for example, A company with critical services might want a backup AS just in case the primal encountered a complete failure.
* Transit AS: a provider of internet access to other AS. The transit AS differ by their size and geographical location, factors that affect the relationships between the ASs.

We will try to perform the attack from different ASs and different policies to see if there are any differences between them.

## System description

We will configure several networks of routers and see how the different parameters affects the efficiency of the attack.

The system will consist different ASs which will be connected through routers.

Each AS will contain some of the components described:

* Routers (IOSv) that will route traffic inside the AS between subnets and between ASs.
* Routem (lxc-routem) control plane traffic generator.
* Servers, using as end point units
* Ostinatos (lxc – ostinato) data traffic generator

We will generate attacks using the routem support of BGP protocol, and will see the effect of the attack by the change in the traffic statistics shown by the ostinato.

## Interfaces used by the project

There are several RESTful API that the VIRL environment exposes that we will use in our project:

* Roster: Gives information about the state of the simulation.
* Simengine: Enables running simulations automatically using scripts.
* AutoNetKit: Gives the initial configurations to the components in the simulation.
* UWM: User Workspace Manager, provides a system management dashboard, and documentation
* Routem: control-plane traffic generator: can create custom BGP packets.
* Ostinato: data-plane traffic generator: to test the affects of the BGP packets sent by routem

# 

# Project Planning

* The time table is continuously set in sync with the mentor of the project based on weekly meetings.
* Mile Stones:
* Setting up and running the VIRL environment
  + Including an option to work outside of the Technicon
* Learning to work with VIRL:
  + How to design networks and topologies?
  + How to configure nodes in the network?
  + Ways to interact with the simulated network:
    - Which APIs exist?
    - How to send packets throughout the network?
    - In which ways can we control the nodes?
  + Specifically investigate the above issues regarding BGP
* Simulating a simple BGP usage scenarios
* Simulating several "Black hole" attacks with different scenarios.
* Answering the following questions:
  + How easy is the execution of the attacks from the attackers stand point?
  + What is the radius of influence?
  + How long does it take the attack to propagate?
  + In which situations such attack is impossible?
  + In which scenarios such attack is relatively easy?
  + How easy is it to catch such attack?
  + Is it easy to track the origin of the attack?
  + What measures could the AS manager take to prevent such an attack? Make it more unlikely?
* Risks:
* The VIRL technology hasn't been used in the lab so far, and there is no documentation of BGP attacks being investigated in this environment
* The simulations, especially ones of large networks, use a lot of resources (the real bottle neck is the memory) and the special server that is currently in the lab might not be capable of handling large simulations

# Appendix

* An example routem code excerpt taken from the VIRL documentation website:

