JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

CYBERSPACE SCENARIO SIMULATION FOR CYBER SECURITY TRAINING LEVERAGING DEVOPS PRINICIPLES

[1]Kunal Sadalkar, [2]Velayutham T.

[1]Member Research Staff, [2]Senior Research Staff

[1]Central Research Laboratory,

[1,2]Bharat Electronics Ltd., Bangalore

Abstract : Cyberspace is the fifth domain of warfare after air, soil, water, and space, where the battlefield is not only limited by physical boundaries but also by virtual ones. Critical infrastructures such as energy, water, nuclear, defense, chemical and healthcare sectors are lucrative targets to the state sponsored attackers at the time of geo-political tensions. Cyber armies of the nations are now adopting "Offense is new Defense" strategy to remain one step ahead of attackers. This has led to cyber arm race which is more dynamic, and subject to change drastically with the introduction of new and complex technologies. In such environment, training and learning cyber-warfare strategies is resource intensive and time consuming. It brings challenges to develop essential cyber security skills and remain up to date with the fast moving industry requirements. To understand how security functions for a target system, it requires a simulation of the target environment repeatedly for different scenarios. These scenarios consist of sets of technologies, network architectures, operating systems, protocols, and services. Installing and configuring such scenarios are complex and time consuming process for a security practitioner. This brings us to derive a methodology to simulate customized, modular, repeatable, a network of system by leveraging recent advancements in virtualization technologies and DevOps principles. In this article, the concept has been validated with common enterprise network scenarios having heterogeneous systems automated over virtual platform. Result offers an easy-to-go approach for building cyberspace scenarios that can be utilized to learn and practice cyber operation strategies.

IndexTerms **- Cyber operation, DevOps, Infrastructure as Code, Virtualization, Automation etc.**

I. INTRODUCTION

Internet services have become an integral part of day-to-day activities. Critical infrastructure services such as energy, water, finance, transport, health, logistics, etc. are digitally woven in the recent era. In spite the advantages brought by modern internet technologies, there are a whole new set of problems as well. Digitization not only brought societal changes, opportunities and wealth, but also altered the battlefield, washed out physical boundaries resulting any nation state from anywhere can launch a cyber-attack against another and cause immense damage without ever firing a bullet shot. The example of such cyber operation would be like disrupting take-off and landing of airplanes from an airfield in conflicted area. This could disrupt and shuts down enemy's flight control system without bombing on runways which leads to a lot of damage and casualties. Therefore, it has become a necessary for nation's defense arsenal to have both defensive and offensive cyber capabilities.

Cyber warfare is a relatively new type of "war" that transcends the traditional way of declaring war. Nations have already invested more than decade ago though the war has not officially started. As per survey, 140 countries across the world have a cyber-operation development programs, making the battle-space much larger and more diverse [4]. China is conducting wide spread APT attacks to steal intellectual property from other countries [8]. US has allocated $647 million budget in the financial year 2018 to build 6200 military and civilian personnel. North Korea's Lazarus APT group stolen US$81 million from Bangladesh central bank using offensive cyber techniques [6]. Israel used cyber offensive operation to temporarily trick Syria's air defense allowing fighter jets to enter Syria unnoticed [10]. In 2016 US presidential election, Russia used range of information warfare techniques to influence the electoral result [7]. It has also accused in engaging in cyber warfare activities against its neighboring countries, such as malfunctioning power stations in Ukraine and attacking government websites in Estonia and Georgia [11]. In October 2019, India announced that North Korean malware designed for data extraction had been identified in the networks of a Kundan-Kulam nuclear power plant [9].

Cyber warfare is termed as the employment of cyber capabilities to achieve objectives in or through cyber space. Cyber-attack is termed as an offensive cyber operation that is reasonably expected to cause injury or death to persons or damage or destruction to objects. Conventionally, a cyber-operation is time consuming process in which intelligence about the target is gathered well in advance. Based on acquired intelligence, multiple attack scenarios are planned. The preferred cyberspace operation plan requires

testing in virtual scenario and finally, tools, technique and procedures (TTPs) needs to be formulated for the successful operation. Therefore, in order to learn and test cyber operation skills, a security researcher needs to simulate different scenarios repeatedly. The scenario generation may include diverse type of technologies, protocols, network topologies, operating systems etc. The processes of setting up such scenarios are time-consuming and error-prone. This may involves repeatable actions of downloading, installing and configuring operating systems, applications and other required services. This requires a high degree of technical ability to create and takes significant time to plan and implement. A similar pattern of problems is found in software development and deployment lifecycles. To address this problem DevOps (Development Operations) culture is evolved where the developer community leverages virtualization and automation technology to recreate a repeatable, sharable, quick development environment.

DevOps is an evolved IT culture in context of system oriented approach that emphasizes on rapid IT service delivery through adoption of lean automation practices and agile methods [13]. It bridges gap between organizations and development teams to deliver and manage life cycle of the software products efficiently. The common problem in software development lifecycle process used to be migrating software application from development environment to operational environment due to differences in the software stack used in both the environment. Developers used to develop a module and pass on to the operational engineer to figure it out how to configure and integrate into existing infrastructure resulting into massive cost in terms of time and money. To overcome this problem, a novel DevOps process has been innovated. In this process, developers began utilizing recent advancements in virtualization and automation technology to reduce cost of software deployment and maintenance.

In this paper, Section II explores requirement and assumptions domain; section III introduces common DevOps tools and technology, section IV focuses on orchestration of sample virtual enterprise scenario using DevOps principle along with achieved results. Finally paper is concluded in section V.

II. REQUIREMENT & ASSUMPTIONS

Security professionals required to develop technical skills on near realistic scenarios. Testing and learning cyber-attacks directly on operational networks is dangerous for security practitioners. Therefore, the scenarios need to be simulated in a controlled environment to test the attack vectors before performing a real-world exercise. It controls the level of risk and consequences of making mistakes, which are often part of the training process. Besides this fundamental requirement, the solution must exhibit the following design characteristics:

- ***Inexpensive***: The infrastructure and maintenance cost of scenario simulation to be inexpensive. The target scenarios are created and destroyed more often. Resource reusability to be adopted for overall cost reduction.
- ***Repeatable***: The scenarios could get damaged during practicing cyber operation attacks. Therefore, the design must provide repeatable scenario generations which have capability to restore previous known stable state.
- ***Easy to manage***: The solution must be easy to manage. Automate manual tedious tasks using automation technology.
- ***Easy to share***: The solution must be portable and easy to share among security practitioners.

Easy to adapt: The system must have provision to adapt for incremental updates of underlying software components. Addition or deletion of the node is to be seamless.

III. TOOLS & TECHNOLOGIES

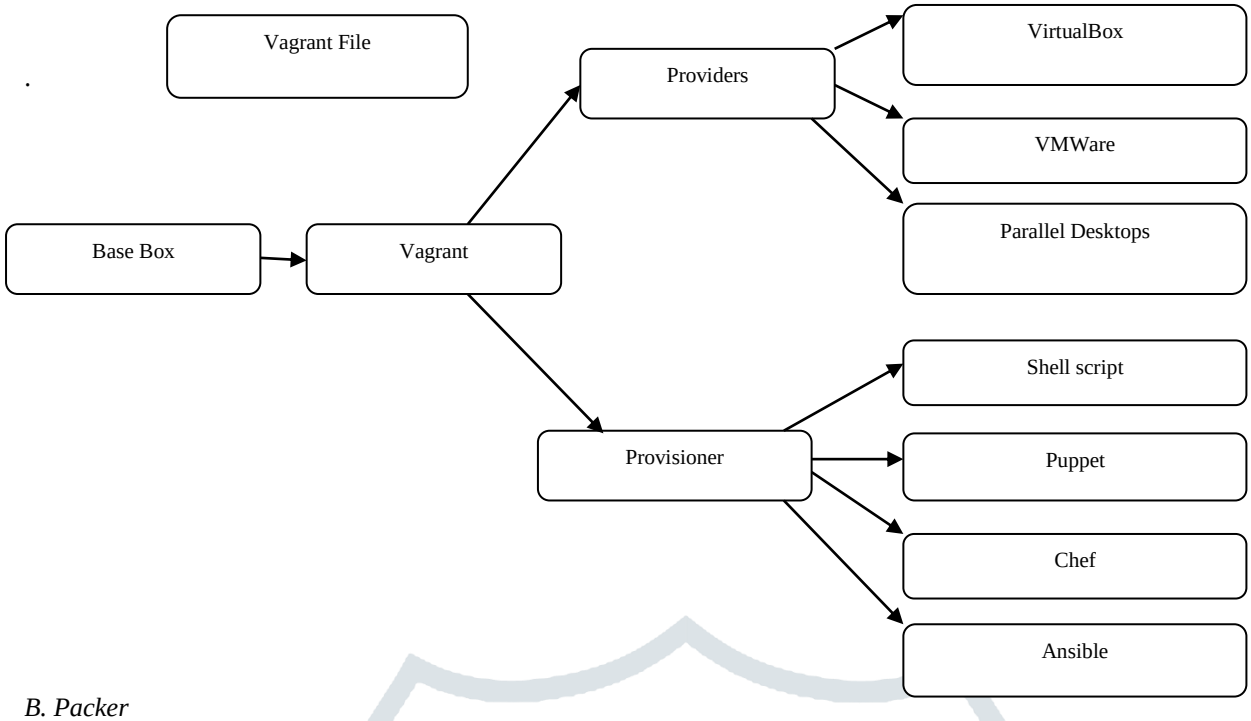
Following section of the paper describes most popular software tools and technology used for virtualization and automation in DevOps culture.

A. Vagrant:

It is an open source application developed by HashiCorp[5] allows to configure, automate and deploy virtual machines on a given virtualized platform using Infrastructure as a Code (IaaC). It provides workflow that mainly focuses on automation and lowering the overall environmental setup time. These workflows are generally defined with a Ruby based configuration files facilitating configurable, reproducible and portable environment. Instead of manually configuring required packages and tools, one can codify in these configuration file called Vagrantfile and run the command Vagrant Up to boot full setup.
Vagrant's design goal is to speed up creation of virtual environments. This property of the system automation process can be leveraged by security practitioners for setting up target cyber security scenarios. Vagrant uses following sub-components to make it generic and supported by industry standard virtualization platforms.

- ***Provisioners:*** These are the tools in Vagrant architecture that allows users to customize the configuration of the virtual environments. Puppet[14], Chef[3], Salt[12], Ansible[2]etc. are some examples of commonly used provisioners in Vagrant ecosystem.
- ***Providers:*** These maps to virtualization engine and their APIs. Vagrant provides out-of the box support for VirtualBox, Hyper-V and Docker etc. Vagrant also allows additional providers through plugin mechanism.
- ***Vagrant Box File:*** It is base VM image with metadata added into file. Particularly built for specific providers and can be used by anyone to bring up that machine. This file can be entire server environment pre-configured with services and application. Vagrant utility can pull these pre-configured box file from public cloud as and when needed.

Vagrant File: It is the file with the set of instruction defines how to configure virtual machines, amount of memory to be allocated for each machine, number of network interface cards and their respective configuration details, number of CPUs etc. It also provides ability to configure and install applications with the help of provisioners



B. Packer

It is an open source software developed by Hashicorp[5] to create identical machine images for given a configuration file. It is lightweight, multiplatform and supports all the major operating systems. The images generated by the packer tool are pre-configured with the applications and can run directly in a public cloud such as AWS or in an OpenStack-based private cloud. The same images can also run in desktop virtualization solution such as Virtualbox or VMWare [17].

IV. ORCHESTRATING VIRTUAL SCENARIOS

For validating concept, we considered a scenario which commonly found in typical enterprise environment. This scenario uses different technologies, operating systems, protocols and services. Thus it requires formulating different attacking approach by the security practitioners depending on the defined target scenario.

Five nodes with single virtual switch network scenario has been constructed and automated using Vagrant tool. The underlying VMWare ESXi bare-metal hyper-wiser platform is running on powerful hardware with 64 GB RAM and E5 series Xeon CPU architecture. Nodes consist of Windows 2019 server, Windows 10 and Windows 7 OS platforms as well as web and mail servers running on Ubuntu Linux. Open source version of pfSense used as a virtual switch to route network traffic among the nodes. This simulates typical simple enterprise network scenario with a firewall at place. Another VM with Kali operating system has been simulated in virtual environment logically residing outside the network firewall. This easy to go approach helps security professionals to reduce the effort required for building scenario which can be utilized to learn and practice offensive strategies.

Table 1 Typical enterprise network scenario network configuration

SL No.	**Enterprise network virtual scenario**		
	Machine Name	***System Information***	***IP address***
1.	vSwitch/firewall	pfSense	192.168.1.1
2.	Windows Client1	Win7	192.168.1.2
3.	Windows Client2	Win10	192.168.1.4
4.	Domain Controller (dc)	Windows Server 2012	192.168.1.3
5.	Web server	Apache/Ubuntu 16	192.168.1.5
6.	Mail Server	Postfix/Ubuntu 16	192.168.1.6

Figure 2 depicts typical scenario of windows based enterprise network. In this particular scenario, security researcher is creating target enterprise network with windows based environment. The near realistic scenario contains heterogeneity in operating systems, architectures, application and services etc. TABLE I shows network configuration of sub-systems of typical windows enterprise network with their names, operating system version and respective IP addresses.

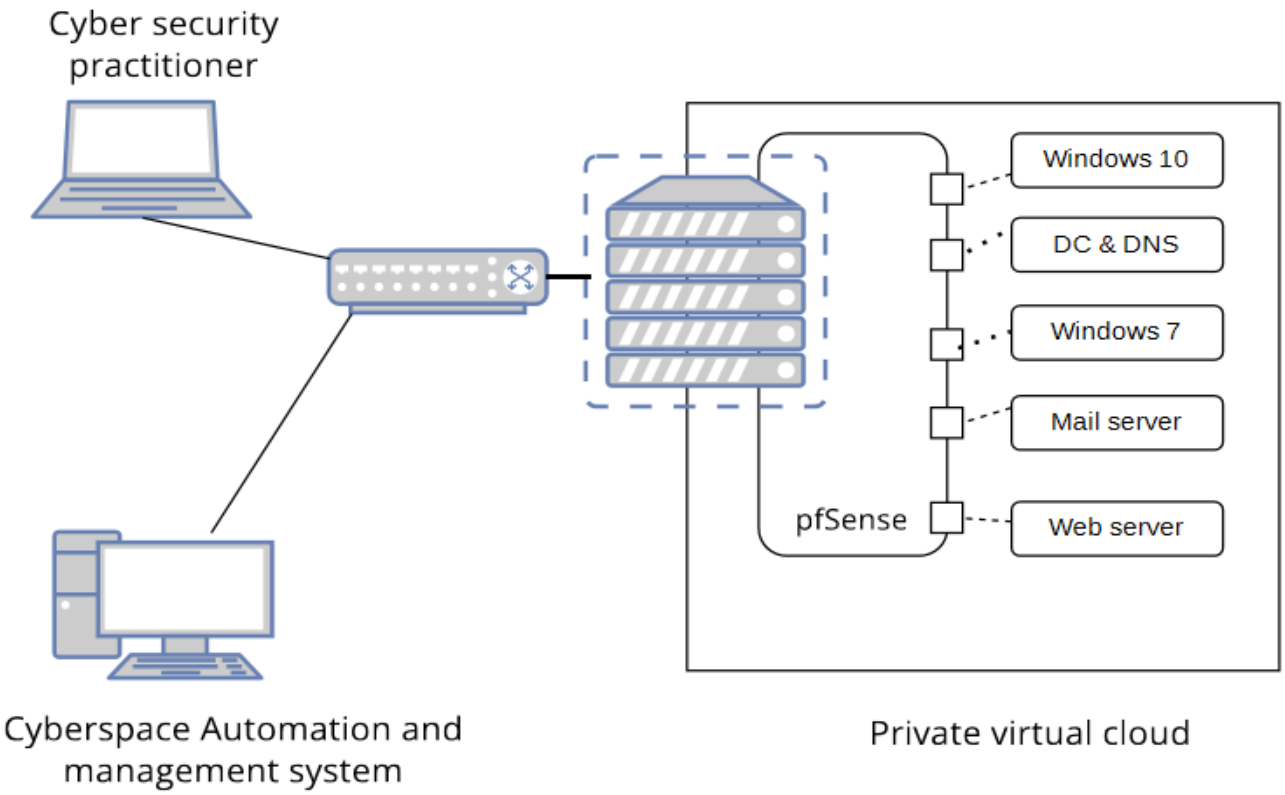


Fig 2. Network topology of typical enterprise network scenario

Pseudo code of vagrant file snippet for automating typical windows enterprise scenario on virtual platform is given in figure 3. Vagrant allows to manage virtual machines through Ruby based scripting language. In this target scenario, there are six virtual machines defined with their resource requirements and network configurations. Each virtual machine copies its template OS from box file, configures network information and executes custom scripts using various provisioners. Here, we are making use of shell scripts as provisioners which gets executed once base OS has been installed. These scripts automate post installation steps such as installing specific version of web server or domain control, configuring respective parameters, template user creations etc. Furthermore, the script is executed using "vagrant up" command to instantiate individual virtual machines and simulates required enterprise scenario. A cyber security practitioner can make use of this scenario to practice defensive or offensive strategies, demonstrate their research, security best practices or cyber exercise for use and participate.

```
forvm = "vSwitch"  {
      define_vm = vSwitch
      define_basebox=  pfSense
      define_hostname  = "vSwitch"
      define_network_type = "dhcp"
      define_provider = "vmware_esxi"
      define_memory = '1024'
}
forvm = "webServer"  {
      define_vm = webServer
      define_basebox=  ubuntu/trusty64
      define_hostname  = "webserver"
      define_network_type = "dhcp"
      define_provider = "vmware_esxi"
      define_memory = '1024'
      define_provision = "script.sh"
}
forvm = "mailServer"  {
      define_vm = mailServer
      define_basebox=  ubuntu/trusty64
      define_hostname  = " mailServer "
      define_network_type = "dhcp"
      define_provider = "vmware_esxi"
      define_memory = '1024'
      define_provision = "script.sh"
}
```

```
forvm = "DC"  {
      define_vm = domain_controller
      define_basebox=  windows2019
      define_hostname  = " DC "
      define_network_type = "dhcp"
      define_provider = "vmware_esxi"
      define_memory = '4096'
      define_numofcpu = '2'
      define_provision = "script.ps"
}
forvm = "client1"  {
      define_vm = client1
      define_basebox=  Windows7
      define_hostname  = "Client1"
      define_network_type = "dhcp"
      define_provider = "vmware_esxi"
      define_memory = '2048'
      define_numofcpu = '2'
      define_provision = "script.ps"
}
forvm = "client2"  {
      define_vm = client2
      define_basebox=  Windows10
      define_hostname  = "Client2"
      define_network_type = "dhcp"
      define_provider = "vmware_esxi"
      define_memory = '2048'
      define_numofcpu = '2'
      define_provision = "script.ps"
}
```

Fig 3. Pseudo-code for automating heterogenious enterprise network scenario

VI. CONCLUSION

Today's connected world, the threat of cyberspace operations have become more realistic than ever. To cope up with this new emerging threat, cyber security professionals must acquire offensive and defensive skills. Learning and exercising cyber operation skill is one of the challenging activities, as the professionals may not always have access to the realistic target environments. Manual target scenario simulation for cyber security research is time consuming and error prone. Target scenarios may get damaged during offensive practices or need to bring back to original state more frequently. However, the need of repeatable, updatable and dynamically reconfigurable cannot be easily achieved through manual process.

In this paper, we derived conceptual approach for automating target scenario using DevOps principles. These target scenarios can be simulated in controlled environment leveraging advances in virtualization and automation technology. This agile approach allows us to dynamically program a realistic target scenario with the required number of instances, underlying operating systems, and specific versions of applications and their network configurations. For validating concept, we automated target scenario with a sample windows enterprise environment containing active directory, DHCP, mail, DNS and web server. The result shows, once initial configuration of the target scenario is ready, the amount of time and cost requires for spinning and updating successive scenario has been drastically reduced or negligible. In few scenarios, the target systems are corrupted due to system memory corruption which was easily restored to stable known condition. Elementary these target scenarios are ruby configuration files; they can be easily shared among other security practitioners to experiment and learn.

In future, research will be extended to derive generic template based framework for multiple cyberspace scenarios with the inclusion of SCADA, IoT, and Wireless security nodes

REFERENCES

[1] Krukowski and I. Kale. Virtual classroom. In *Advanced Learning Technologies, 2001. Proceedings. IEEE International Conference on, 2001*.

[2] Ansible is Simple IT Automation. *http://www.ansible.com*.

[3] Chef IT Automation for Speed and Awesomeness https://www.chef.io/chef

[4] Cyber Warfare -- Reasons Why Israel Leads The Chargehttps://www.forbes.com/sites/christopherskroupa/2017/09/07/cyber-warfare-reasons-why-israel-leads-the-charge

[5] Development Environments Made Easy. https://www.vagrantup.com

[6] https://www.bbc.com/news/stories-57520169

[7] https://www.justice.gov/archives/sco/file/1373816/download

[8] https://www.justice.gov/opa/pr/two-chinese-hackers-associated-ministry-state-security-charged-global-computer-intrusion

[9] https://www.washingtonpost.com/politics/2019/11/04/an-indian-nuclear-power-plant-suffered-cyberattack-heres-what-you-need-know

[10] https://www.wired.com/2009/11/mossad-hack/

[11] https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid/

[12] Intelligent, event-driven IT automation. *https://www.saltstack.com*

[13] KaChing Chan and M. Martin. *"An integrated virtual and physical network infrastructure for a networking laboratory*". In Computer Science Education (ICCSE), 2012 7th International Conference on, pages 1433– 1436, July 2012

[14] Make software discovery, management, and delivery automatic and pervasive with Puppet. *https://puppet.com/*

[15] N. Childers et al., "*Organizing Large Scale Hacking Competitions*," in Proceedings of the 7th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, Berlin, Heidelberg, 2010, pp. 132–152

[16] OpenStack: An open source cloud solution *https://www.openstack.org/*

[17] Packer:https://www.packer.io/intro /

[18] R. Beuran, K. Chinen, Y. Tan, Y. Shinoda. "*Towards Effective Cybersecurity Education and Training."* Research report, IS-RR-2016-003, Japan Advanced Institute of Science and Technology (JAIST), Ishikawa, Japan, October 2016

[19] V. Tam, A. Yi, and E.Y. Lam. "*Building an interactive simulator on a cloud computing platform to enhance students"* understanding of computer systems. In *Advanced Learning Technologies (ICALT), 2013 IEEE 13th International Conference on, pages 154–155, July 201.3*

[20] Z. C. Schreuders and E. Butterfield, "*Gamification for Teaching and Learning Computer Security in Higher Education*," in 2016 *USENIX Workshop on Advances in Security Education (ASE 16)*, Austin, TX, 2016.