ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 3, pp: (137-140), Month: July - September 2023, Available at: www.researchpublish.com

MITIGATING THREATS THROUGH APPLICATION PROGRAMMING INTERFACE INSPECTION AND MANIPULATION IN VMWARE VCENTER

Badi Salah[1], Badar Qahtani[2], Mohammed Jugaiman[3], Saleh Falah[4]

Saudi Aramco, Dhahran, Saudi Arabia

DOI: https://doi.org/10.5281/zenodo.8315071

Published Date: 04-September-2023

Abstract: **To ensure the safety and security of VMWare vCenter instances, this research paper aims to enhance the security by building upon the work known as the "Secure by Design Private Cloud Infrastructure." To enhance the security presented by the paper, this study dives deeper into securing vCenter a critical component within the VMWare private cloud ecosystem by leveraging API gateway technology. The aim is to intercept, analyse, and manipulate Application Program Interface commands (API) to filter malicious attacks on vCenter. This will provide a seamless way to secure vCenter, the main orchestrator of the private VMWare cloud, to mitigate the risk of an attacker ultimately break and disrupt critical applications hosted within the private cloud.**

Keywords: **private cloud, information technology, cyber security, vcenter, API.**

I. INTRODUCTION

In recent years, the landscape of enterprise IT infrastructure has undergone a key transformation as organizations transition from traditional hardware-centric data centers to Software Defined Data Centres (SDDCs) powered by private and public cloud technologies. This paradigm shift brings various benefits, ranging from enhanced cost efficiency and flexibility to seamless business continuity (IBM, 2019). These changes greatly elevated the criticality of management systems such as VMware vCenter, which orchestrates and manages virtualized environments within the cloud ecosystem while being a main integration point for many solutions. This made these central management systems a main target for malicious actors to the point where enterprises must strive to protect them.

The prior paper, "Secure by Design Private Cloud Infrastructure," aimed to dissect the resiliency of the private cloud infrastructure to threat actors. Major design weaknesses were apparent after performing a detailed assessment and analysis. Therefore, these findings required design improvements to secure the private cloud which ultimately protect the data within. Building upon these design improvements, this paper aims to provide further enhance security to the central cloud management which adds to the security of the solution as a whole. Specifically, this paper investigates the use of an Application Programming Interface (API) gateway to intercept, inspect, and potentially manipulate commands sent to cloud central management (vCenter). This unique approach aims not only to mitigate potential threats but also to gain visibility and control over intercepted commands that affect the central management and, thus, the private cloud as a whole.

II. MOTIVATION

Due to the criticality of the cloud central management, which can operate and access all data and services on the cloud, a rigorous security approach is expected from any enterprise that deems IT services critical to its operations. Many mitigations and security considerations already exist, including the ones discussed in the previous paper, yet this unique way will

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 3, pp: (137-140), Month: July - September 2023, Available at: www.researchpublish.com

provide an extra layer of protection to safeguard the environment specifically for the central management. This methodology also provides further insights and statistics into commands received by the central management, which can allow advanced malicious detection methodology in addition to a way to analyse and optimize the performance of the central management.

III. METHODOLOGY

The idea is to place an intermediate machine running an algorithm that can intercept, inspect, and manipulate commands sent to the vCenter. To explore this method, a Burp proxy was placed before a test vCenter to intercept traffic:

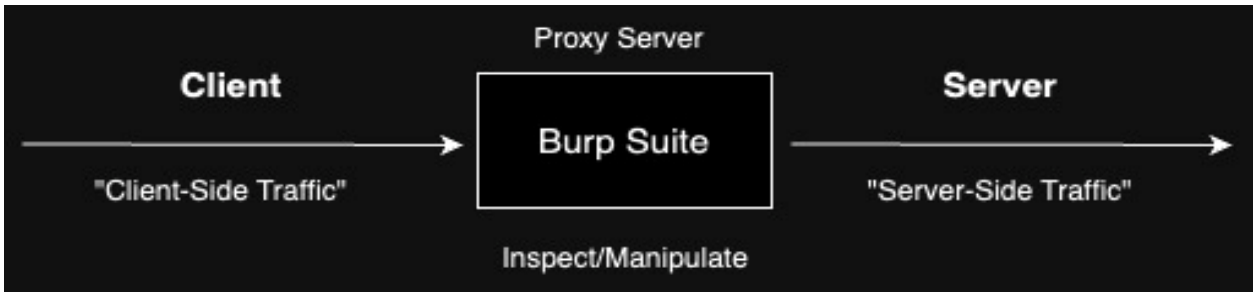

Figure 1: Architecture

```
POST /ui/mutation/validateSpec?propertyObjectType=com.vmware.vsphere.client.provisioning.spec.OvfReferencesValidationSpec HTTP/1.1
Host: vcenter-main.
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://vcenter-main. domain .com/ui/
Content-Type: application/json;charset=utf-8
webClientSessionId: 0b7567e5-0bba-4581-92b3-a5798d51c295
X-VSPHERE-UI-XSRF-TOKEN: 76a4a69f-5982-4cd5-b690-413d539a4883
Content-Length: 299615
Connection: close
Cookie: VSPHERE-UI-XSRF-TOKEN=76a4a69f-5982-4cd5-b690-413d539a4883; JSESSIONID=388CB5BD8524C15EDBC12FE8347CFDE6; VSPHERE-USERNAME=Administrator%40VSPHERE.LOCAL;
VSPHERE-CLIENT-SESSION-INDEX=_e5e1b28a915b4d088e384eb8d350a441; XSRF-TOKEN=KlDH4tRfOF7hkE9eDrRJOUdymH7HuaR4;
appliance-ui-sessionid=NmUyYmQwYzktYmY5Zi00ODM3LWI3N2YtMjQ5ODZiMDMzOGY0; CastleSessionvsphere.local=_1b33f7c2f99efd4b9f851a21d0d55dff

{
  "descriptorContents": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<ovf:Envelope xmlns:ovf=\"http://schemas.dmtf.org/ovf/envelope/1\"
```

Figure 2: Burp interception

Based on the observed traffic, it is possible to perform a content rewrite before passing the commands on to vCenter. This will help to filter unsafe vCenter operations from any user, even legitimate ones, thus mitigating the threat of malicious actors. For instance, all commands done by the local admin user (administrator@vsphere.local) can be filtered to force cloud administrators to use their own users instead of the general one for auditing purposes. Another use case for this is to block datastore browse operations cloud, leading to data leakage if abused:

```
POST /ui/authorization/privileges/entities/ HTTP/1.1
Host: vcenter-main. domain .com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://vcenter-main. domain .com/ui/
Content-Type: application/json;charset=utf-8
webClientSessionId: a02ad866-097f-4c18-b1ee-c55eafb9ff4e
X-VSPHERE-UI-XSRF-TOKEN: fbe03a9e-67f8-4924-ad25-58a678074a80
Content-Length: 170
Connection: close
Cookie: VSPHERE-UI-XSRF-TOKEN=fbe03a9e-67f8-4924-ad25-58a678074a80; JSESSIONID=73D4D431405AFFA4C5EED99906C2CAF1; VSPHERE-USERNAME=Administrator%40VSPHERE.LOCAL;
VSPHERE-CLIENT-SESSION-INDEX=_4ee15ff41f2da940c096ff2595ad9b7d; XSRF-TOKEN=KlDH4tRfOF7hkE9eDrRJOUdymH7HuaR4;
appliance-ui-sessionid=NmUyYmQwYzktYmY5Zi00ODM3LWI3N2YtMjQ5ODZiMDMzOGY0; CastleSessionvsphere.local=_83846be9c473d986c2f2ce2aae047166

{
  "objectIds": [
    "urn:vmomi:Datastore:datastore-10:da712671-e3b5-4bdb-966f-202432d6507e"
  ],
  "properties": [
    "Datastore.Browse",
    "Datastore.Config"
  ]
}
```

Figure 3: Burp manipulation

Such a system can be managed by a specialized security team with its specialized updatable signature database. This allows different authority entities to secure vCenter, thus further enhancing the security of the private cloud.

IV. DESIGN PROS AND CONS

This solution would imply some strengths and drawbacks as follows:

Strengths

- Provide further analysis on what commands vCenter is facing, allowing even machine learning for optimization. Furthermore, analysis can be used to optimize the performance of the vCenter itself by filtering out redundant or unneeded commands.
- An extra layer of defense capable of filtering out pre-configured scenarios to stop malicious actors.
- It can be used to enforce security compliance in many instances for example filtering the use of the local administrator.
- Provide role-based authority to allow security entities further control over the commands sent to vCenter.
- Simple design with a requirement of a single gateway to filter requests.
- Transparent to the vCenter and users.

Drawbacks

- Due to the nature of intercepting traffic, performance could be impacted if the gateway could not keep up with the commands sent.
- Maintaining the gateway adds extra responsibility to maintain the compatibility and configurations of the solution, thus requiring expert workforce.
- This could cause significant confusion and add to the complexity of diagnosing issues.
- Vendor support may consider such setup as unsupported.
- Administrators may resist such setup as it limits their ability to operate and diagnose vCenter.
- If the gateway itself is compromised, then all commands and authority tokens would be exposed to an attacker.

V. CONCLUSION

The outcome of this experimental research has proven that an API gateway proxy can mitigate the risk of malicious actors targeting vCenter. Furthermore, it can help enforce security compliance and aid in performance tuning of the central management. This proposal can also be applicable to another critical control planes given the proper configurations. Drawbacks of this solution are expected, with performance being the top drawback that requires further analysis. Another major drawback would be the administration overhead, such as maintaining the system and configurations, especially with software upgrades/updates impacting the API calls.

REFERENCES

[1] (J.R.)Winkler, V. (2011). Evaluating Cloud Security: An Information Security Framework. In V. (J.R.)Winkler, Securing the Cloud (pp. 233-252). Elsevier Inc.

[2] Cappuccio, D. (2013, July 1). Software Defined Data Centers – Hype or Reality? Retrieved from Gartner Blog Network: https://blogs.gartner.com/david_cappuccio/2013/07/01/software-defined-data-center

[3] Dimitrios, Z., & Dimitrios, L. (2012, March). Addressing cloud computing security issues. Retrieved from Science Direct: https://www.sciencedirect.com/science/article/pii/S0167739X10002554

[4] Faatz, D. (2018, March 12). Best Practices for Cloud Security. Retrieved from Carnegie Mellon University: https://insights.sei.cmu.edu/sei_blog/2018/03/best-practices-for-cloud-security.html

[5] Hange, M. (2011). Security Recommendations for Cloud Computing Providers. Retrieved from German Federal Office for Information Security.

[6] How to audit the cloud. (2019). Retrieved from ICAEW.

[7] IBM. (2019). Cloud computing: A complete guide. Retrieved from IBM Cloud website: https://www.ibm.com/cloud/learn/cloud-computing

[8] National Institute of Standards and Technology. (2012, Sep). Guide for Conducting Risk Assessments.

[9] Rouse, M. (2018, Fev). cloud SLA (cloud service-level agreement). Retrieved from Tech Target: https://searchstorage.techtarget.com/definition/cloud-storage-SLA

[10] Suby, M. (2014, July). Best Practice Security in a Cloud-Enabled World.

[11] Badi Salah, Hasan Ahmadi, Mohammed Fadlalla, Mohammed Jugaiman, “Secure By Design Private Cloud Infrastructure”, ISSN 2348-1196 (2021, Nov)