Developing a Remedy to Volumetric DoS and SAN Based Attacks on Datacenters with Virtual Machine Migration and Traffic Engineering

Cyrus Majd
1/22/18
Honors Advanced Research

Abstract

Today's datacenters host thousands of virtual machines (VMs), which are commonly connected to a centralized storage infrastructure via the Storage Area Network (SAN). While centralized storage systems are both cost-efficient and convenient, they pose a potential security vulnerability and can easily be exploited by cyber criminals to inflict mass harm on a datacenter. This is most commonly done with a cyber criminal conducting some variant of a storage-based Denial of Service attack (DoS) that would effectively render critical storage services inaccessible to other VMs and devices.

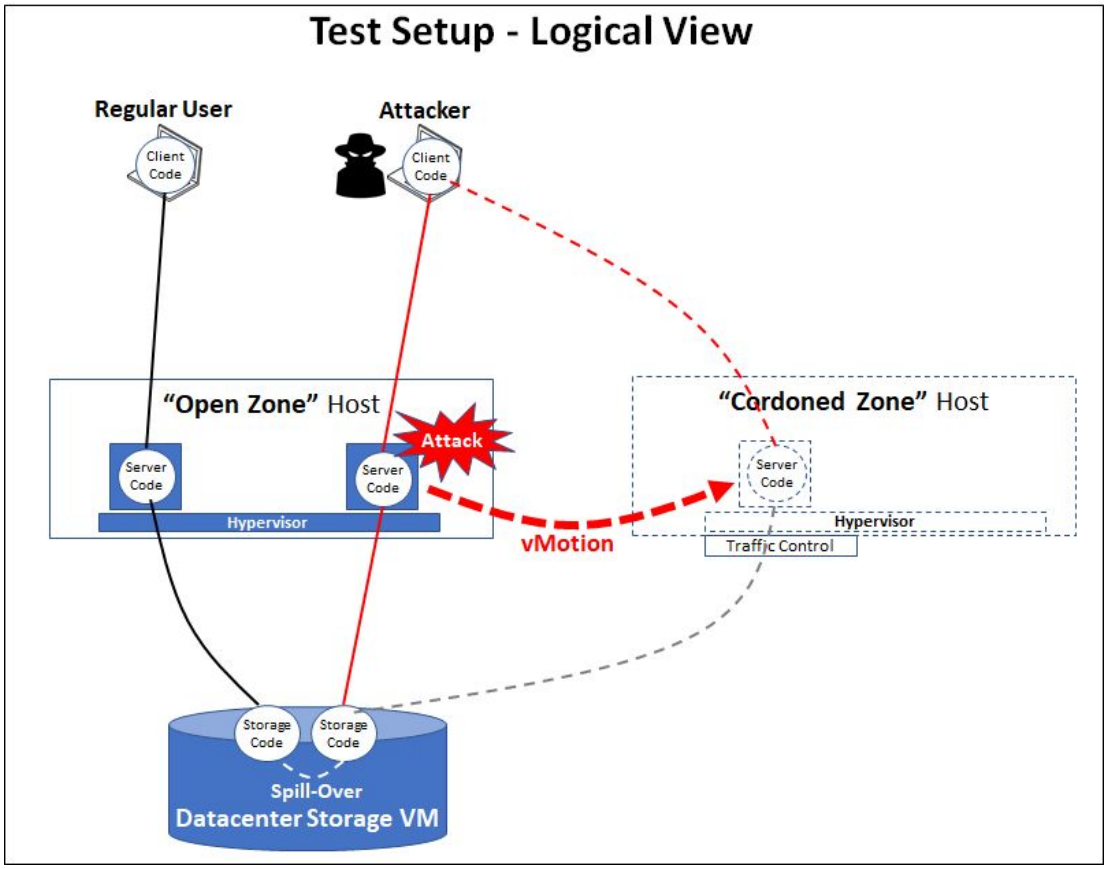
In this research project, a method was devised to mitigate the effects of such attacks by using virtual machine migration to transfer an attacked VM from a host in an "Open Zone" to a new host in a "Cordoned Zone". When attacked VMs are migrated to the Cordoned Zone, the shared storage system will be relieved from DoS attack traffic, and in doing so, the impact of such an attack is drastically diminished. This technique is used to create a new "intrusion remediation system", which could complement existing datacenter intrusion detection & prevention systems.

Introduction

Large datacenters constitute much of the critical IT infrastructures in today's modern societies. Many vital services in our daily lives depend on the proper operation of the cloud. Unfortunately, such large datacenters are a primary target for cyber criminals. Currently, a number of intrusion detection & prevention systems are used to detect and fend off cyber attacks, but these techniques rely on the static and complex configuration of firewalls and security patches. Often such systems are not capable of preventing new kinds of attacks, and are merely used to identify and record attack signatures to prevent them from reoccurring in future. Cyber criminals frequently change their attack signatures, and thus bypass most datacenter security systems with ease once they have found any kind of software vulnerability. In an age of high-speed internet and cloud computing, there is a need for a more agile and real-time remediation system to fend off cyber attacks.

Datacenters often share expensive infrastructure among many VMs to make their services more affordable for mass consumption. However, the sharing of core infrastructure and services can constitute security threats and become critical attack targets for cyber criminals. Whenever such shared resources are compromised, attackers can steal sensitive information and deny others access to key infrastructure vital for normal functionality.

This research focused on investigating a method for detecting attacks on shared storage systems in real-time and accurately measuring an attack's impact on other service's performance. A topology of the network used in this research project is shown below.



The "Cordoned Zone" is a virtualized environment designed to have a limited and seperate connection to the internet and to the SAN. Once a regular VM is attacked, it will be migrated from the "Open Zone" to the Cordoned Zone, where it will inherit the network properties of its new environment. Because this research project used VMware's hypervisors, the vCenter Appliance was used to migrate the VMs between the hosts with a technique called "vMotion". vMotion is typically used for power efficiency rather than cybersecurity, which exemplifies how software is not limited only by its intended purpose. A simple way to understand the Cordoned Zone is to imagine it as a quarantine room for "sick patients" (VMs). Once the "sick patients" are no longer dangerous (once the attack is remedified and ultimately put to an end), the "patients" are free to return (in this case, to the Open Zone).

Methods:

The research consisted of three parts. Those were

1. Setting up the test infrastructure,
2. Writing the code for simulating cyber attacks, and finally
3. Performing the tests, collecting data, and analyzing the results

Test Infrastructure

The infrastructure used in this research consists of the following:

1. The "user client" laptop that runs the user client code and simulates a regular user
2. The "attacker client" laptop that runs the attacker client code and launches cyber attacks on shared infrastructure
3. The “Open Zone” physical host that is used to host all server VMs that are serving user traffic and hence are open for public access. That includes supporting the “user server” VM and the “attacked server” VM.
    1. The “user server” VM that connects to the "user client" laptop and runs server software.
        i. The “server software” is a code that performs the simple job of receiving web requests from its user client and forwarding them to the “storage”(SAN VM) for data retrieval. The “user server” runs in the datacenter on a “Open Zone” physical host that could initially be shared with the “attacked server”.
    2. The “attacked server” VM that connects to the "attacker client" laptop and runs the same server software as the “user server”.
        i. The job of the “attacked server software” is to receive web requests from its attacker client and forward them to the storage VM for data retrieval. The “attacked server” runs in the datacenter on the same “Open Zone” physical host that is shared with the “user server”.
            1. Since this physical host is shared between both the “user server” and the “attacked server”, it represents a security threat and can be used as a target by the attacker to disrupt the user processes.
4. The “Cordoned Zone” physical host that is used to temporarily host the “attacked server”, once an attack is detected.
    1. This Cordoned Zone physical host plays the role of a “hospital” for “infected” VMs. Anything hosted in this Zone inherits bandwidth limitation properties and is essentially isolated from the “user server”.
5. The storage VM that runs on a separate physical host and connects to both the “user server” and the “attacked server”.
    1. By connecting to both servers, the storage VM constitutes a shared infrastructure and hence presents a security thread to the datacenter, meaning it could be targeted by an attack to disrupt user processes. The physical host of the storage is kept separate to avoid any interdependencies to other computers. The storage VM runs two independent storage software codes, one serving the “user server” and the other serving the “attacked server”. These software modules perform the simple task of taking incoming requests from their corresponding servers and responding to each by a random number that simulates data retrieval from the database. The speed of the random number generator (RNG) used in

the storage can be controlled to simulate different query times on the database. This way the response time for each request can be controlled during the experiments.

6. The vCenter server is a software that controls the entire virtualized infrastructure and can be used to perform vMotion. The vCenter is connected to both servers and can move the “attacked server” from the “Open-Zone” physical host to the “Cordoned-Zone” physical host, once an attack is detected.
    1. vCenter supports live migration, meaning that VMs are moved from the Open Zone to the Cordoned-Zone while staying online and receiving live traffic. The entire vMotion action is very fast and takes just a few milliseconds. Therefore the user does not perceive noticeable disruption to its traffic.
    2. Once the attack ceases, vCenter can migrate the VM back to the Open Zone to resume regular user traffic. This way, in a large datacenter, one could imagine many VMs to be shipped back and forth between the open and the cordoned zones as attacks take place during operational hours. There would be no need for static pre-configurations and the system can react dynamically to threads in real-time. This solution is also arguably more efficient than shutting the entire datacenter down or rerouting all datacenter traffic to a different location, which is currently the best “remedy” for modern attacks of this scale.
7. A VM hosting an open-source operating system called WANatronic. This VM has the power to limit any traffic that passes through it. By configuring the Cordoned Zone to use the WANatronic VM as a proxy, an attacker’s malicious requests could be routed through the network-limiting software and be nullified.

Software/Code

The software used in this research consisted of the following modules. These software modules were all developed to simulate attacks, run multiple tests, and collect many data points.

- The "User Client" software is a Java client that places web calls to a server. The code is shown below:

```
for(int j=0; j<numberOfCalls; j++){
    TimeUnit.MILLISECONDS.sleep(hitPeriod);
                                    clientTime_in = System.currentTimeMillis();
   String response= client.target("http://10.0.0.47:8080/attackcode/webapi/server/1/?rngCount=" + rngCount).request
                                      (MediaType.TEXT_PLAIN).get(String.class);
  clientTime_out = System.currentTimeMillis();
}
```

- The "Attacker Client" code runs on the attacker laptop and is used to simulate storage request attacks:

```
public static void main(String[] args) throws JSONException {
                                final Client client = ClientBuilder.newClient();
                                int attackPeriod;
```

```
                    final ScheduledExecutorService executorService =
Executors.newSingleThreadScheduledExecutor();
                    executorService.scheduleAtFixedRate(new Runnable() {
                    @Override
                    public void run() {
    String response =client.target("http://10.0.0.48:8080/AttackCodes/webapi/ attackedserver/2/
      ?rngCount=25000000").request(MediaType.TEXT_PLAIN).get(String.class);
                      }
                    }, 0, attackPeriod, TimeUnit.MILLISECONDS);
                }
```

- The server code that runs on both the user server and the attacked server is shown below. Its job is to route requests from the client and forward them to the storage VM:

```
public String getJson(@PathParam("attackCount") int attackCount,  @QueryParam("rngCount") int rngCount) throws
JSONException, InterruptedException {
                    Long serverTime_in = System.currentTimeMillis();
                    String response = client.target("http://10.0.0.37:8082/attackcode/webapi/storage/"
                            + attackCount + "/?rngCount=" +
rngCount).request(MediaType.TEXT_PLAIN).get(String.class);
                    Long serverTime_out = System.currentTimeMillis();
                    return jsonObject.toString();
                }
```

- This is the storage code, which runs on the storage VM. Its job is to receive the forwarded requests from the servers and respond to each request with a random number. The random numbers are then multiplied by 50 so as to generate numbers larger than 1. The duration of the random number computation is controlled with a for-loop in the User Client code and is infinite in the Attacker Client, since the attacker is constantly “straining” the storage VM without end.

```
public String getJson(@PathParam("attackCount") int attackCount, @QueryParam("rngCount") int rngCount) throws
JSONException, InterruptedException {
                    Long storageTime_in = System.currentTimeMillis();
                    for (int i = 0; i <= rngCount; i++) { randomNumber = (int) (Math.random() * 50);}
                    Long storageTime_out = System.currentTimeMillis();
                    return jsonObject.toString();
                }
```

Test Procedure

The test procedure consisted of the following steps:

1. Measurement of the control group
    1. The control group represents a case where there is no cyber attack. The User Client places a group of 60 web requests to a server, which in turn forwards the requests to the storage VM.
    2. The storage VM responds to each request by computing a random number and returning it back to the client via the intermediate server.

3. The storage code measures the time span between the arrival of each request to the departure of its response. This timespan is called the "Storage Query Time" and it is used as the independent variable for the tests.
4. The independent variable is changed from ca. 50ms to 300ms total computation time. This is to demonstrate any kind of trend observable from requests of different computational intensities.
5. The client receives a machine timestamp between the start and end of each call and computes the request delay. This difference in time calculates the total duration of one "round trip" between the client to the storage and back, as opposed to the computation time calculated by the storage VM.
6. The user client also computes the total length of the elapsed time between the beginning of the first call to the end of the last (60th) call. We call this timespan the total length of all individual calls and use it as the dependent variable. The total length of all calls represents the delay that a user would experience when accessing a datacenter VMs.
7. Figure 1 (Control Group without Attack) presents the measured data.The diagram below shows the results.
8. Figure 4 represents the data on a graph as "No Cyber Attack".

2. Measurement of the first experimental group
    1. The first experimental group represents how much the "Length of Call" for the user is delayed because of the attacker's traffic on the shared storage VM.
    2. In this test, the user client places 60 web calls to the storage while the attack client places consecutive calls to the same shared storage.
    3. The storage has to serve both, the user's workload and the attacker's workload. Hence the attacker's workload delays the user workload.
    4. The larger the number of requests the storage VM has to process, the longer the delay that the user workload will experience due to the attack.
    5. The data for this test is shown in Figure 2.
    6. The results are shown in Figure 4 as "With Cyber Attack".
    7. As expected, a doubling of the delay experienced by the user due to the attack was observed.
    8. *This test proves that a shared infrastructure such as the storage VM constitutes a "weak point" with regard to security and can be used by attackers to disrupt other users workloads in cloud computing datacenters.*
3. Measurement of the second experimental group after vMotion
    1. In this last test, we perform vMotion and migrate the attacked VM from the Open Zone to the Cordoned Zone and repeat the tests of the previous experimental group.
    2. In this test, the user client again places 60 web calls to the storage while both the Query time and the total call duration are measured.
    3. The data is presented in Figure 3 under the label "Experimental vMotion Group"
    4. Figure 4 shows the results (cyber attack remedied by vMotion)
    5. The results show that after the VM is moved to the Cordoned Zone, the user traffic experience reduced delays due to attack.

i. *These results prove that vMotion can be indeed used to create a intrusion remediation system. The migration of attacked VMs from the Open Zone to the Cordoned Zone can fend off the impact of a cyber attack on shared infrastructure.*

6. **The results prove that this technique can be used to remediate the impact of cyber attacks on shared resources, while still maintaining high oversubscription of those resources.**

Data Analysis and Discussions:

Control Group (without Attack)	58	113	156	210	263	313	331
	2.942	5.767	7.577	10.915	14.228	15.257	15.872
	2.708	5.623	7.7	10.364	13.711	15.357	16.382
	2.737	5.635	8.355	10.841	13.43	15.768	15.666
	2.886	5.696	7.745	9.986	12.711	16.998	16.487
	3.171	5.584	7.833	10.862	12.913	17.201	17.301
	2.708	5.696	7.902	10.082	13.479	15.256	16.075
	3.559	5.619	7.638	9.832	12.209	15.051	17.202
	2.775	5.548	7.654	10.562	12.365	15.154	16.365
	2.74	5.475	7.845	10.427	12.229	15.359	16.299

	58	113	156	210	263	313	331
MIN	2.708	5.475	7.577	9.832	12.209	15.051	15.666
Q1	2.737	5.584	7.654	10.082	12.365	15.256	16.075
MEDIAN	2.775	5.623	7.745	10.427	12.913	15.357	16.365
Q3	2.942	5.696	7.845	10.841	13.479	15.768	16.487
MAX	3.559	5.767	8.355	10.915	14.228	17.201	17.301
Box 1 - hidden	2.737	5.584	7.654	10.082	12.365	15.256	16.075
Box 2 = lower	0.038	0.039	0.091	0.345	0.548	0.101	0.29
No Cyber Attack	0.167	0.073	0.1	0.414	0.566	0.411	0.122
Whisker Top	0.617	0.071	0.51	0.074	0.749	1.433	0.814
Whisker Bottom	0.029	0.109	0.077	0.25	0.156	0.205	0.409

Figure 1 (Above): Control group data (Regular user requesting responses from a VM without cyber attacks)

Experimental Group (with Attack)	58	113	156	210	263	313	331
	4.371	10.112	15.974	20.683	22.016	25.802	30.724
	4.734	10.094	15.913	20.682	22.32	26.109	31.748
	4.503	10.714	16.236	21.195	21.301	26.212	31.221
	4.582	10.499	16.28	20.786	22.932	26.009	31.227
	4.325	9.56	15.667	21.298	22.219	25.906	31.129
	4.348	10.061	17.201	21.604	21.809	26.829	31.23
	4.309	9.661	15.973	20.478	22.425	26.298	31.025
	4.501	9.92	15.77	20.785	22.015	26.64	30.82
	4.412	9.898	15.867	20.58	23.449	26.519	30.616

	58	113	156	210	263	313	331
MIN	4.309	9.56	15.667	20.478	21.301	25.802	30.616
Q1	4.348	9.898	15.867	20.682	22.015	26.009	30.82
MEDIAN	4.412	10.061	15.973	20.785	22.219	26.212	31.129
Q3	4.503	10.112	16.236	21.195	22.425	26.519	31.227
MAX	4.734	10.714	17.201	21.604	23.449	26.829	31.748
Box 1 - hidden	1.406	3.006	6.176	9.514	7.681	9.319	11.161
Box 2 = lower	0.064	0.163	0.106	0.103	0.204	0.203	0.309
With Cyber Attack	0.091	0.051	0.263	0.41	0.206	0.307	0.098
Whisker Top	0.231	0.602	0.965	0.409	1.024	0.31	0.521
Whisker Bottom	0.039	0.338	0.2	0.204	0.714	0.207	0.204

Figure 2 (Above): Experimental Group data (Attacker and user sharing the same storage VM. Attack is not remediated)

Experimental vMotion Group (With Attack and vMotion Remedy)	58	113	156	210	263	313	331
		6.892	8.902	10.817	13.414	15.155	20.479
		6.985	8.086	11.168	14.641	17.099	19.659
		7.409	9.691	11.313	13.618	16.69	19.25
		6.61	7.841	10.546	14.334	16.104	20.069
		5.841	10.475	10.903	14.232	16.456	19.25
		6.381	10.335	11.155	14.335	17.713	18.225
		5.898	8.564	10.712	13.925	16.177	18.943
		5.944	7.856	10.466	13.822	15.768	17.303
		6.498	8.59	11.284	13.412	16.215	17.817

	58	113	156	210	263	313	331
MIN		5.841	7.841	10.466	13.412	15.155	17.303
Q1		5.944	8.086	10.712	13.618	16.104	18.225
MEDIAN		6.498	8.59	10.903	13.925	16.215	19.25
Q3		6.892	9.691	11.168	14.334	16.69	19.659
MAX		7.409	10.475	11.313	14.641	17.713	20.479
Box 1 - hidden		0.248	0.241	-0.129	0.139	0.336	1.738
Box 2 = lower		0.554	0.504	0.191	0.307	0.111	1.025
Cyber Attack remedied by vMotion		0.394	1.101	0.265	0.409	0.475	0.409

Figure 3 (Above): Experimental Group data (Attacker and user sharing the same storage VM. Attack is remediated with vMotion)

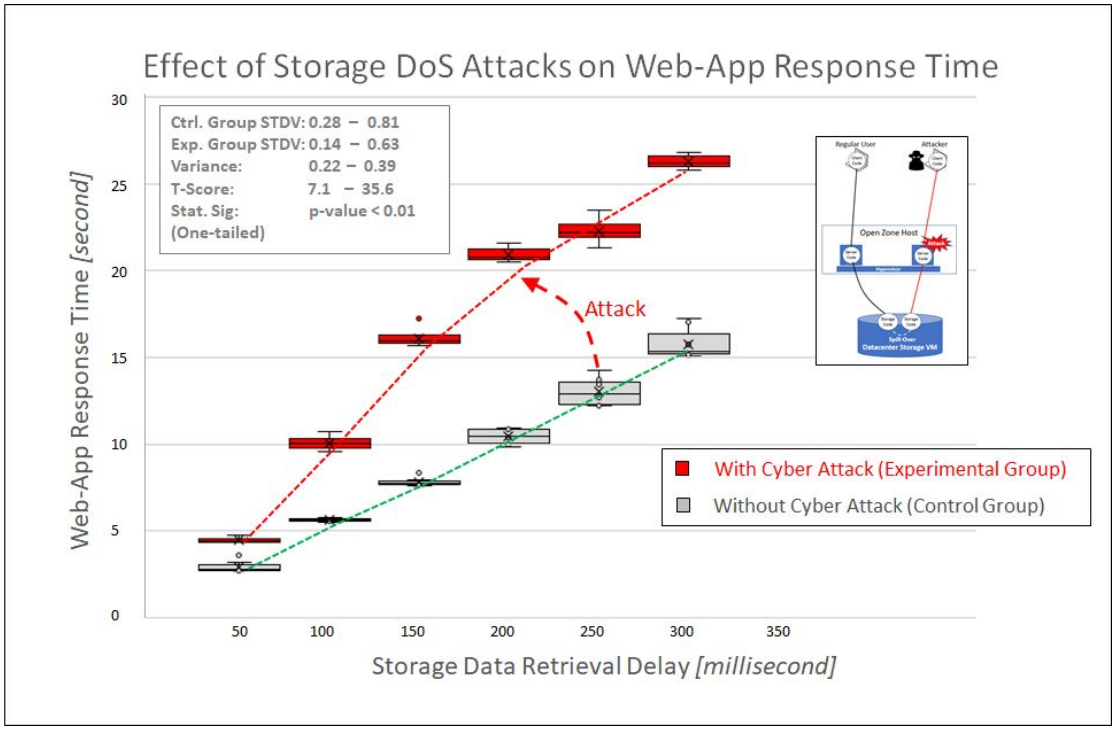


Figure 4 (Above): Total Call Length (in seconds) as a function of the Storage VM's Response Time (in milliseconds) without remediation.

Statistical Analysis on Figure 3:

1. Number of tests for each curve:	6 x 10 x 50 tests
2. STDV of the means for Control Group:	from 0.28 sec to 0.81 sec
3. STDV of the means for Experimental Group:	from 0.14 sec to 0.63 sec
4. Variance between the two Groups:	from 0.22 to 0.39
5. Degrees of Freedom of two Groups:	16 samples
6. The t-Score:	from 7.1 to 35.6
7. The t-test for Stat. Significance (one-tailed):	p-Value < 0.01

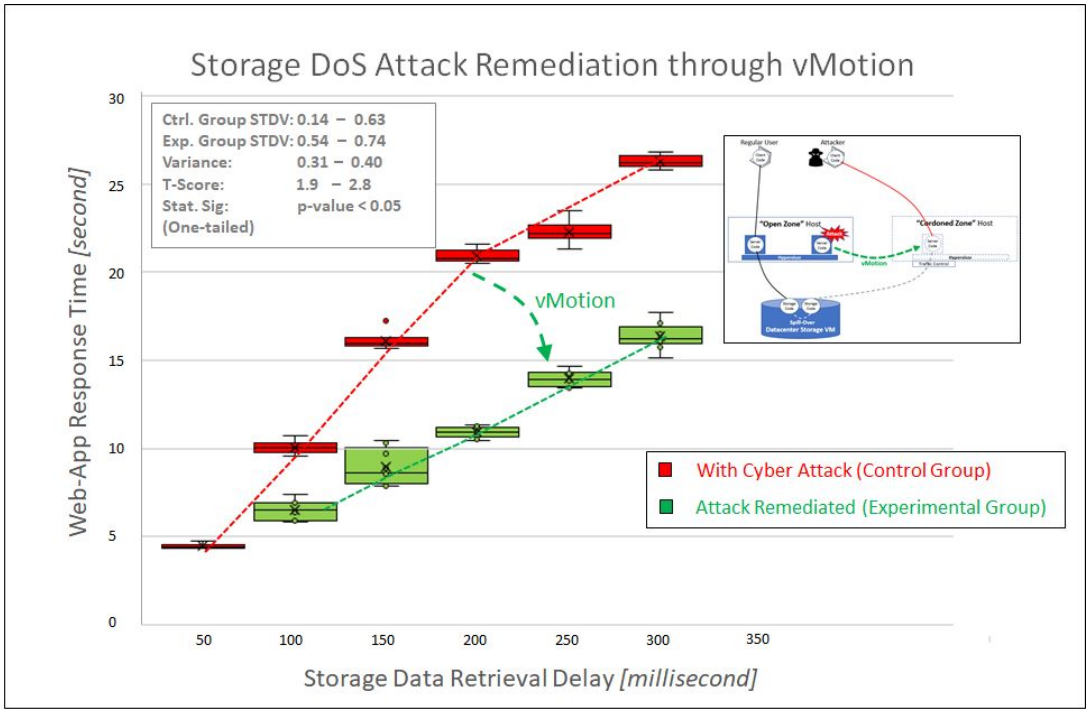


Figure 4 (Above): Total Call Length (in seconds) as a function of the Storage VM's Response Time (in milliseconds) with remediation.

Statistical Analysis on Figure 4:

1. Number of tests for each curve:	6 x 10 x 50 tests
2. STDV of the means for Control Group:	from 0.14 sec to 0.63 sec
3. STDV of the means for Experimental Group:	from 0.54 sec to 0.74 sec
4. Variance between the two Groups:	from 0.31 to 0.40
5. Degrees of Freedom of two Groups:	16 samples
6. The t-Score:	from 1.9 to 2.8
7. The t-test for Stat. Significance (one-tailed):	p-Value < 0.05

The results of the tests are shown in the diagram above. The data shows that:

- The control group consists of the data center workload without any attacks and acts as reference for measuring the effectiveness of attacks on data center performance. The results in Figure 1 show that the end-to-end length of calls increase from ca. 2ms to 15ms as the storage response time is increased from 50ms to 300ms in steps of 50ms. This behavior was expected.
- The results in Figure 4 show that the total length of the call transactions for the regular data center users increases as long as it is exposed to an attack. Comparing these results with those of the control group shows that the user workload can be slowed down by up to 100% due to an attack. This observation is aligned with my hypothesis that attacking the same resources the user utilizes will significantly slow down the user's process completion time.

- The results in Figure 4 show that that the total length of call transactions for the regular data center users during the attack is drastically reduced when the attacked server was migrated from the Open Zone to the Cordoned Zone using vMotion. The reduction of the attack impact is in this case 90%, and it can be controlled depending how strongly we restrict the traffic flow to and from the cordoned zone. These results prove the concept that the combination of vMotion and a "cordoned zone" can be effective in building a data center remediation solution.

Conclusions

This research proves that when shared infrastructure in a data center is exposed to denial of service attacks, the performance of all user applications that share that infrastructure could be negatively impacted. In fact, the results show that an attacker's workload can slow down a user's workload by up to 100%.

Furthermore, this research proves that virtual machine migration can be used to remediate cyber attacks by moving attacked VMs to a resource-limited zone (the Cordoned Zone). The test results show that by doing so the impact of attacker on user traffic can be eliminated. vMotion, an intended power saving tool, is discovered to also be an excellent intrusion remediation tool when utilized in the correct way. The procedure is risk-free, since VMs can be migrated back to the Open Zone when the attack has ceased or is finally blocked by an intrusion prevention system.

The insights gained through the tests proved the concept of a "datacenter Intrusion Remediation System". Data centers currently use commercial Intrusion Detection and Prevention Systems that are weak in eliminating initial attacks and rely on static configuration and frequent patches. However, the intrusion remediation system demonstrated in this project can complement the shortcomings of existing security systems and make datacenters more flexible and secure.