Studying the ARP Spoofing Attack Effect on SDN Networks

Dr. BoushraMaala*
Mohammed Abd Al-Hameed**

(Received 11 / 11 / 2019. Accepted 8 / 3 / 2020)

□ ABSTRACT □

The mapping of Layer 3 (IP) to Layer 2 (MAC) addresses is a key service in IP networks, and is achieved via the Address Resolution Protocol (ARP) protocol in IPv4. Due to its stateless nature and lake of authentication, ARP is an easy goal to spoofing attacks, which can enable Denial of Service (DoS) or Man-in-the-Middle (MIM) attacks.
In this search, we discuss the problem of ARP spoofing in the context of Software Defined Networks (SDNs). We studied important parameters such as throughput, delay and the availability of the network. Results showed that ARP spoofing attacks was able to make a negative effects on network performance.

Keywords: SDN, ARP spoofing, Security, DoS, MIM.

* **Associate Professor, Department of Communication and Electronics, Factually of Mechanical and Electrical Engineering, Tishreen University, Lattakia, Syria.** boushra.maala@gmail.com
** **Postgraduate Student, Department of Communication and Electronics, Factually of Mechanical and Electrical Engineering, Tishreen University, Lattakia, Syria** mohammedabdolhamed589@gmail.com

دراسة تأثير هجوم خداع ARPعلى الشبكات المعرفة بالبرمجيات SDN

د. بشرى معلا •

محمد عبد الحميد••

(تاريخ الإيداع 11 / 11 / 2019 . قُبِل للنشر في 8/ 3 / 2020)

□ ملخّص □

تعد عملية التخطيط بين عناوين الطبقة الثالثة (IP) والطبقة الثانية (MAC) المفتاح الرئيسي في شبكات IP، وذلك من خلال بروتوكول دقة العناوين ARP في شبكات IPv4. بما أن هذا البروتوكول غير مستقر ولا يستخدم أي آلية للمصادقة فإنه يعد هدفاً سهلاً لهجمات الخداع. قد تؤدي هذه الهجمات بدورها إلى هجمات أعقد مثل هجوم الرجل في المنتصف MIM وهجوم حجب الخدمة DoS.

ناقشنا في هذا البحث مشكلة هجوم خداع البروتوكول ARP من خلال الدراسة في سياق الشبكات المعرفة بالبرمجيات SDN. دُرست مجموعة من البارامترات الهامة مثل الإنتاجية والتأخير وتوافرية الشبكة. بينت النتائج أن هجمات خداع البروتوكول ARP قادرة على التأثير سلباً على أداء الشبكة.

الكلمات المفتاحية :الشبكات المعرفة بالبرمجيات، خداع ARP، الأمن، هجوم حجب الخدمة، هجوم الرجل في المنتصف.

• أستاذ مساعد، قسم هندسة الاتصالات والالكترونيات، كلية الهندسة الميكانيكية والكهربائية، جامعة تشرين، اللاذقية، سورية، boushra.maala@gmail.com

•• طالب ماجستير، قسم هندسة الاتصالات والالكترونيات، كلية الهندسة الميكانيكية والكهربائية، جامعة تشرين، اللاذقية، سورية، mohammedabdolhamed589@gmail.com

مقدّمة:

اعتبرت شبكات SDN الحل الواعد للشبكات المستقبلية. إن الفكرة الأساسية لشبكات الـــSDN هي في فصل مستوى التحكم عن مستوى البيانات، والسماح بالإدارة الفعالة والمرنة وتشغيل الشبكة من خلال البرمجيات. تؤدي الأجهزة (المبدلات والموجهات) في مستوى البيانات تمرير الرزم اعتماداً على قوانين منصبَّة من قبل المتحكم. يرى المتحكم في مستوى التحكم كامل البنية التحتية ويؤمن منصة فعَّالة ومرنة لتطبيق خدمات وتطبيقات شبكات مختلفة. إضافة إلى ذلك،تسمح شبكات SDN بالإدارة المركزية للتحكم العكسي واتخاذ قرارات أفضل من خلال نظرة عامة عن الشبكة والمعلومات المتبادلة بين الطبقات[1].يعد البروتوكول ARP من البروتوكولات الشهيرة المستخدمة ضمن شبكاتSDN،حيث يمتلك كل جهاز ضمن الشبكة زوجاً من العناوين IP/MAC. عندما تريد عقدة ما إرسال البيانات لعقدة أخرى، تحتاج كلاً من هذين العنوانين للعقدة الهدف[2]. يعد البروتوكول ARP بروتوكولاً غير مستقر حيث يمكن لأية عقدة أن تتعامل مع رسالة ARP على أنها رسالة صحيحة علماً أنها قد تأتي من عقد مهاجمة. الأمر الذي قد يؤدي بدوره إلى حصول هجوم خداع بروتوكول ARP (ARP Spoofing). يعد هذا الهجوم من أخطر الهجمات التي قد تتعرض لها الشبكة حيث من الممكن أن تنتج عنه هجمات أعقد مثل هجوم الرجل في المنتصف Man–in–the–Middle (MIM) أو هجوم حجب الخدمة Denial of Service (DoS) وغيرها من الهجمات [3].

أهمية البحث وأهدافه:

تأتي أهمية هذا البحث من حيث أنه يتناول موضوعاً حديثاً نسبياً. تتركز الدراسات لإيجاد تعديلات إما في التطبيقات التي تتعامل معها شبكات SDN أو التعديل في البروتوكول ARP من أجل تحصينه ضد الهجمات.

يهدف هذا البحث إلى تحليل هجوم الرجل في المنتصف MIM المعتمد على هجوم خداع ARP، وهذا الجانب غاية في الأهمية خاصةً في مثل هكذا نوع من الشبكات، وذلك لأنه في حال تمكن المهاجم من الاستماع إلى الاتصال بين أي مضيفين فإنه سيكون قادراً على الوصول إلى الرزم التي يرسلها المتحكم المسؤول عن الشبكة، وبالنتيجة سيؤدي إلى السيطرة على كامل الشبكة.

طرائق البحث ومواده:

طبقت سيناريوهات المحاكاة باستخدام متحكم Floodlight[4] مدمج مع محاكي mininet2.2 باستخدام نظام Ubuntu14.0[5]. كما استخدمت مجموعة من الأدوات التي يتيحها هذا المحاكي من أجل قياس البارامترات في الشبكة حيث درست الإنتاجية throughput من خلال الأداة iperf[6] ونفذ هجوم خداع ARP باستخدام الأداة DSniff[7]، وأيضاَ اعتمدت الأداة Cbench[8]من أجل دراسة التأخير (الذي يمثل الزمن اللازم لانتقال رزمة بين المضيفين ووصول رسالة الرد acknowledgment وهو ما يسمى بزمن الذهاب والعودة RoundTripTime) والإنتاجية في المتحكم (والتي تمثل معدل البتات التي تصل بشكل سليم خلال واحدة الزمن) .

اعتمد هجوم رجل في المنتصف إذ سيتم العمل على ثلاثة أنواع من أشهر الطوبولوجيات في شبكات SDN وهي الطوبولوجيا ذات الشبكة الشجرية [9] والطوبولوجيا ذات المبدل الوحيد [10] والطوبولوجيا الخطية [11].

1. الشبكات المعرفة بالبرمجيات SDN

تعد منظمة التشبيك المفتوح Open Networking Foundation (ONF)[12] مؤسسة غير ربحية مسؤولة عن التطوير ووضع المعايير والتسويق لشبكات SDN. لقد قدمت هذه المؤسسة أكثر تعريف منطقي لهذه الشبكات بأنها بنية شبكة جديدة بحيث أن التحكم بالشبكة مفصول عن التمرير كما أنها قابلة للبرمجة مباشرة.

تتلخص فكرة شبكاتSDN بأنها تؤمن قابلية البرمجة من خلال الفصل بين مستويي التحكم والتطبيقات. أي تتضمن شبكات SDN أجهزة شبكة بسيطة قابلة للبرمجة بدلاً من زيادة تعقيد أجهزة الشبكة كما هو الحال في التشبيك الفعال. علاوةً على ذلك، تؤمن شبكاتSDN فصل المستويين من خلال تصميم بنية الشبكة. بهذا التصميم،سيكون ممكناً التحكم بالشبكة بشكل منفصل ضمن مستوى التحكم دون التأثير على تدفق المعطيات.

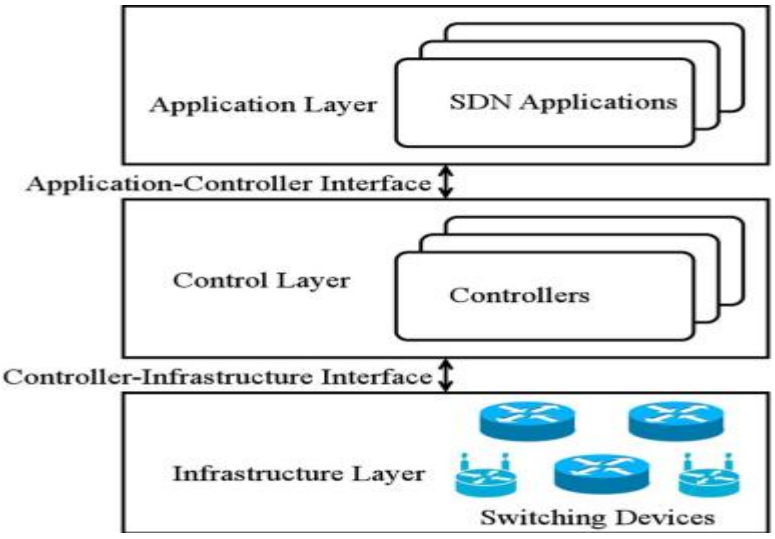
يمكن أن يؤخذ ذكاء الشبكة من أجهزة التبديل ويوضع في المتحكمات. في نفس الوقت، فإن أجهزة التبديل يمكن التحكم بها من خلال البرمجيات. يؤمن الفصل بين مستويي التحكم والمعطيات ليس فقط بيئة بسيطة قابلة للبرمجة، وإنما يعطي حرية أكبر للبرمجيات الخارجية لتعريف سلوك الشبكة.

1.1 البنية الطبقية لشبكات SDN

لقد اقترحت الـــONF نموذجاً مرجعياً لشبكات SDN كما هو مبين في **.(Error! Unknown switch argument.** يتألف هذا النموذج من ثلاث طبقات [12,1] هي طبقة البنية التحتية وطبقة التحكم وطبقة التطبيقات متوضعة فوق بعضها البعض.

1. طبقة البنية التحتية (Infrastructure Layer):

تتألف من أجهزة التبديل (مبدلات، موجهات...) في مستوى البيانات. وظيفة أجهزة التبديل هذه مضاعفة، فهي تعد المسؤولة عن تجميع حالة الشبكة وتخزينها مؤقتاً ضمن أجهزة محلية وإرسالها إلى المتحكم. قد تتضمن حالة الشبكة معلومات مثل طوبولوجيا الشبكة،وحالة الحركية، واستخدامات الشبكة، كما أنها المسؤولة عن معالجة الرزم المرتكزة على قوانين موضوعة من قبل المتحكم.



الشكل (1): البنية الطبقية لشبكات SDN

2. **طبقة التحكم (Control Layer):**

تصل ما بين طبقة البنية التحتية وطبقة التطبيقات من خلال واجهتيها. من أجل الوصلة بالاتجاه السفلي (تسمى واجهة جنوبية) والتي تتفاعل مع طبقة البنية التحتية فإنها تحدد الوظائف للمتحكمات من أجل الوصول إلى الوظائف المقدمة من قبل أجهزة التبديل. قد تتضمن هذه الوظائف تقارير عن الشبكة وقوانين تمرير الرزم. من أجل الوصلة بالاتجاه العلوي(تسمى واجهة شمالية) والتي تتفاعل مع طبقة التطبيقات فإنها تؤمن نقاط الوصول للخدمة بصيغ مختلفة، على سبيل المثال، واجهة التطبيق القابلة للبرمجة Application Programming Interface (API). يمكن لتطبيقات SDN الوصول إلى التقارير عن معلومات حالة الشبكة المقدمة من أجهزة التبديل من خلال هذه الـ APIs،وجعل قرارات النظام مرتكزة على هذه المعلومات، وتحميل هذه القرارات من خلال وضع قوانين التوجيه للرزم لأجهزة التبديل أيضاً من خلال هذه الواجهات. من الممكن وجود أكثر من متحكم بغرض زيادة الإدارة للشبكة،وهذا يستدعي الحاجة إلى واجهات اتصال شرق–غرب (east–west) بين المتحكمات من أجل مشاركة معلومات الشبكة والتنسيق فيما بينها لاتخاذ القرارات.

3. **طبقة التطبيقات (Application Layer):**

تحتوي على تطبيقات SDN المصممة لتلبي رغبات المستخدمين. من خلال المنصات القابلة للبرمجة المزودة من قبل طبقة التحكم، تكون تطبيقات SDN قادرة على الوصول إلى أجهزة التبديل في طبقة البنية التحتية والتحكم بها.

2.1 المتحكمات في شبكات SDN

يوجد العديد من المتحكمات المستخدمة في شبكات SDN والجدول (1) يبين الفرق بين أشهر أنواع المتحكمات [12,13].

جدول 1: مقارنة بين أنواع المتحكمات

الاسم	لغة البرمجة المستخدمة	المنظمة المسؤولة عنه	الخصائص
NOX/POX	C++/Python	Stanford	أول متحكم في شبكات SDN
Beacon	Java	Stanford	يدعم العمليات المقادة بالأحداث ومزود بواجهة WebGUI
Floodlight	Java	Big Switch	سهولة التطوير والكثير من التوثيق
Ryu	Python	NTT laboratories	مدعوم من قبل OpenStack
Open Daylight	Java	Linux Foundation	مزود بـــ WebGUI وواجهات قابلة للبرمجة

3.1 المتحكم Floodlight

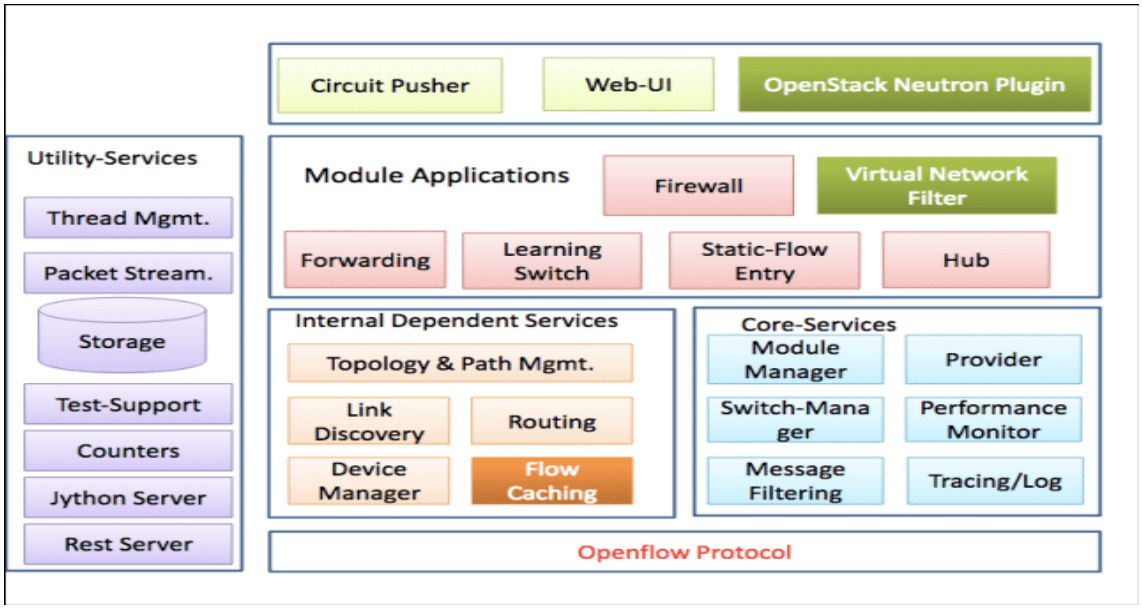
هو متحكم مفتوح المصدر وذو واجهة Web، مبني بلغة Java وأحد أكثر أنواع متحكمات SDN شيوعاً[4] . تقسم بنية المتحكم Floodlight إلى مجموعة من الوحدات حيث يوجد وحدة للإدارة ووحدة للطوبولوجيا وموازنة الحمل.

يمكن النفاذ لهذه الوحدات من خلال واجهة المستخدم الرسومية المعتمدة على الإنترنت WebGUI. يقدم المتحكم واجهة قابلة للبرمجة بحيث يمكن للتطبيقات الوصول إلى المتحكم والشبكة. يستخدم البروتوكول Link Layer Discovery Protocol (LLDP)[14]لاكتشاف طوبولوجيا الشبكة. يتعامل مع الأجهزة في الشبكة ويحول رسائل Open Flow[15]إلى أحداث.

يعتمد على ثلاث واجهات أساسية للأحداث هي Switch Listenerو Device Listener و Message Listener.

تستخدم الواجهة Switch Listener لاستقبال الإشعارات عندما يتم اتصال/قطع اتصال مبدل بالإنترنت أو في حال تغير حالة المنفذ. بينما تستخدم الواجهة Device Listener لإعطاء إشعارات عندما تتم إضافة، إزالة أو تغيير عنوان IP لجهاز ضمن الشبكة. وتستخدم الواجهة Message Listener عند استقبال رزمة من قبل المتحكم،فعند استقبال رزمة يقوم التطبيق بمعالجتها واتخاذ الإجراء المناسب. يؤمن متحكم Floodlight نوعين من التطبيقات: التفاعلية والاستباقية.

يبين الشكل (2) بنية المتحكم Floodlight[16].



الشكل(2): بنية المتحكم Floodlight

4.1 الأمن في شبكات SDN

إن طبيعة تصميم شبكات SDN جعل الهجمات عليها أكثر خطورةً مقارنةً مع الشبكات التقليدية. رغم ذلك فهي تمتلك بعض المزايا الخاصة [17]. فنذكر من إيجابياتها ما يأتي:

✓ الرقابة الفعالة للحركية غير الطبيعية (Effective monitoring of abnormal traffic):

يملك المتحكم نظرة عامة عن عمل الشبكة والحركية بشكل متزامن، لذا سيكون من السهل ملاحظة السلوك غير الطبيعي في حركية الشبكة والمسبب من قبل المهاجم.

✓ التعامل الزمني مع نقاط الضعف(Timely dealing with vulnerabilities):

من المزايا الهامة لشبكات SDN هي بيئة الشبكة القابلة للبرمجة، حيث أنه في حال اكتشاف هجوم جديد فإنه يمكن للمشغل أن يقوم بإعداد برمجيات جديدة للتحليل والتعامل مع هذه المخاطر دون انتظار زمن حتى يُحدَّث نظام التشغيل أو تُحدَّث برمجيات التطبيق في الأجهزة من قبل المصنع.

من ناحية أُخرى تواجه شبكاتSDN عدة مخاطر:

- ✓ المتحكم الضعيف (Vulnerable controller):

تتركز معظم الوظائف، مثل تجميع معلومات الشبكة وإعداد الشبكة وحسابات التوجيه في متحكم SDN. ففي حال تمكن المهاجم من السيطرة عليه فإنه سيكون قادراً على إصابة الشبكة بشلل من ناحية الخدمات وقد يؤثر على كامل الشبكة المغطاة من قبل المتحكم.

- ✓ المخاطر المسببة من قبل الواجهات القابلة للبرمجة :(Risks caused by open program able Interfaces)

وفقاً لطبيعة هذه الشبكات، فإن شبكات SDN هي الأكثر عرضة للمخاطر الأمنية. أولاً، هذه الشبكات تجعل البرمجيات في متحكم SDN مكشوفة للمهاجمين، وفي النهاية سيمتلك المعلومات الكافية لتشكيل استراتيجية الهجوم. ثانياً، يزود المتحكم طبقة التطبيقات بعدد كبير من الواجهات القابلة للبرمجةلكن قد يسبب هذا المدى من الانفتاح إلى سوء استخدام الواجهات مثل إضافة كود برمجي مزيف كفايروس. لذلك فإن الواجهات المفتوحة لمتحكمات SDN تحتاج إلى التطوير بدقة.

2. بروتوكول دقة العناوين (Address Resolution Protocol)ARP

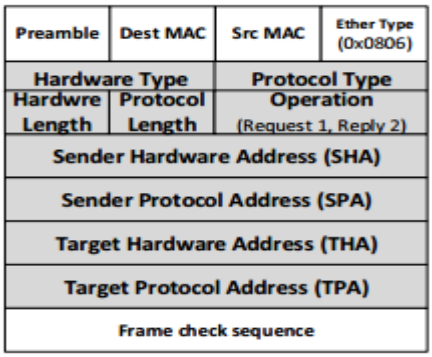
يعد البروتوكول ARP من البروتوكولات الشهيرة المستخدمة ضمن شبكاتSDN، يمتلك كل جهاز ضمن الشبكة زوجاً من العناوين IP/MAC. عندما تريد عقدة ما إرسال البيانات لعقدة أخرى، تحتاج كلاً من هذين العنوانين للعقدة الهدف [2]. يتم تزويد عنوان IP للعقدة الهدف من خلال بعض بروتوكولات الطبقات العليا ضمن نموذج OSI المعياري. نحتاج من أجل إيجاد عنوان الـ MAC إلى بروتوكول دقة العناوين ARP. إن البروتوكول ARP هو بروتوكول طبقة ثانية يقوم بإيجاد عنوان MAC عندما يكون عنوان IP معروفاً. ويحافظ أيضاً جدول ARP على تخطيط العناوين ما بين IP–MAC ضمن الذاكرة المخبئية (cache). والشكل الآتي يبين مثالاً على جداول ARP.

```
Router#show ip arp

                     Age
Protocol  Address          (min)  Hardware Addr   Type   Interface
Internet  172.16.233.229     -   0000.0c59.f892   ARPA   Ethernet0/0
Internet  172.16.233.218     -   0000.0c07.ac00   ARPA   Ethernet0/0
Internet  172.16.168.11      -   0000.0c63.1300   ARPA   Ethernet0/0
Internet  172.16.168.254     9   0000.0c36.6965   ARPA   Ethernet0/0
```

الشكل(3): مثال عن جدول ARP ضمن ذاكرة cache في موجه Cisco

تتألف بنية إطار ARP بشكل عام من الحقول المبينة في الشكل (4).



الشكل(4): بنية إطار ARP

وتكون الحقول الرئيسية لهذا البروتوكول هي:

(Sender Hardware Address (SHA: العنوان الفيزيائي للمرسل MAC.

(Target Hardware Address (THA: العنوان الفيزيائي للهدف MAC.

(Sender Protocol Address (SPA: العنوان المنطقي للمرسل IP.

(Target Protocol Address (TPA: العنوان المنطقي للهدف IP.

عندما تريد عقدة ما أن ترسل بياناتها إلى عقدة أخرى، فإنها تحتاج لإنشاء إطار، يجب أن يحتوي هذا الإطار على عناوين MAC لكل من عقدتي المصدر والهدف. تقوم العقدة المرسلة بفحص جدول ARP الخاص بها لإيجاد عنوان الـــ MAC للهدف. في حال لم يكن عنوان MAC موجوداً تقوم هذه العقدة بإرسال طلب ARP(ARP request) كرسالة بث عام إلى جميع العقد الموجودة ضمن شبكة LAN بحيث يكون حقل عنوان MAC للهدف في طلب ARP هو FF-FF-FF-FF-FF-FF. تستقبل كل العقد ضمن LAN طلب ARP وتتفحص فيما إذا كان عنوان IP ضمن هذا الطلب هو عنوان IP الخاص بهذه العقدة. تقوم العقدة التي يتطابق عنوان IP الخاص بها مع عنوان IP في طلب ARP بإرسال إجابة ARP(ARP reply) إلى العقدة المطلوبة.

1.2 هجوم خداع ARP(ARP Spoofing)

المشكلة الأساسية في البروتوكول ARP أنه يعالج كلَّ طلب أو إجابة(request or reply) بشكل مستقل عن أي اتصال سابق. كنتيجة لذلك،سيقبل المضيف بسهولة المعلومات من رسائل ARP reply دون أن يرسل أي طلب. وبما أنا لبروتوكول ARP لا يملك أية آلية تحقق من المرسل لرزمة ARP سواء طلب أم إجابة، ولا يملك أية آلية للتأكد من صحة المعلومات الموجودة في الرسالة، فإنه من السهل على المهاجم السيطرة على ذاكرة ARP الخاصة بالمضيفين من خلال تخطيط مزيف لعنواني IP-MAC. كل ما يحتاج المهاجم القيام به هو تعديل رسالة الطلب أو الإجابة بحقل SPA خاطئ، فيستقبل المضيف الرسالة ويحدّث جدول ARP الخاص به على أساسها[10].

2.2 المخاطر الناتجة عن هجوم خداع ARP

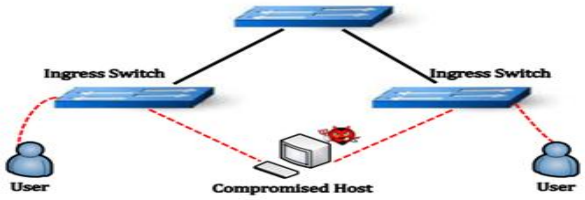
قد تستخدم هجمات خداع ARP كجزء من هجمات كبيرة على الشبكة منها [10]:

1. هجوم حجب الخدمة Denial of Service (DoS): يقوم المهاجم بتزييف جدول ARP لمضيف ما، وبذلك ترسل الرزم المرسلة إلى هذا المضيف إلى المهاجم بدلاً منه وبذلك يتمكن المهاجم من حجب الاتصالات من وإلى المضيف.
2. انتحال شخصية المضيف Host impersonation: بدلاً من إهمال الرزم التي استقبلت، يقوم المهاجم بالرد عليها ويمكن انتحال شخصية أي مضيف في الشبكة.

3. هجوم الرجل في المنتصف Man-in-the-middle (MIM): من خلال خداع مضيفين ضمن الشبكة، إذ يمكن للمهاجم أن يتوضع في منتصف الاتصال بين المضيفين والاستماع إلى الاتصالات بين هذين المضيفين. يكون بذلك المهاجم قادراً على الاستماع إلى الحركية المرسلة في الاتجاهين كليهما. كما يمكن القيام بهذا الهجوم بين مضيف وبوابة العبور gateway في الشبكة من أجل التنصت على الاتصالات الخارجة من الشبكة.

3. هجوم الرجل في المنتصف MIM (Man-in-the-Middle):

يمثل هجوم الرجل في المنتصف كما هو مبين في الشكل(5)،طريقةً كلاسيكيةً لاقتحام الشبكة. إن الفكرة الأساسية له هي من خلال حقن عقدة دخيلة ما بين عقدتي المصدر والهدف بهدف اعتراض معلومات الاتصال والتلاعب بها دون أن يتم كشف ذلك من قبل أي من أطراف الاتصال [18].



الشكل(5): هجوم MIM

من بعض الأمثلة على هجمات الرجل في المنتصف في هذه الشبكات:

1. خداع ARP(ARP Spoofing)
2. خداع DNS(DNS Spoofing)
3. سرقة الجلسة (Session Hijacking)

من أشهر أساليب هجوم الرجل في المنتصف MIM هو خداع ARP، حيث يمكن للمهاجم أن يخدع مضيفين اثنين ضمن الشبكة في نفس الوقت كما هو مبين في **Error! Unknown switch argument.**6)، عندها يتمكن المهاجم من الاستماع بصمت إلى الحركية بين العقدتين. في هذه الحالة، يستطيع المهاجم الوصول إلى معلومات سرية كما يمكن له أن يعدل المعلومات المرسلة.

الشكل(6): هجوم الرجل في المنتصف اعتماداً على البروتوكولARP

في حال نُفذ هجوم الرجل في المنتصف MIM بنجاح ضمن شبكة SDN سيؤدي إلى هجمات أخرى منها هجوم حجب الخدمة Denial of Service (DoS)، بحيث يغرق المتحكم بالرزم على اعتبار أنه أصبح من عقد الشبكة مما يتسبب في حجب المتحكم عن القيام بوظائفه.

يصنف هذا النوع من الهجمات كهجوم مثالي ضد شبكة SDN، حيث يمكن له اعتراض قواعد التمرير الموجهة للمبدل والحصول على التحكم الكامل بالشبكة. وبعد إنجاز ذلك يمكن تطبيق العديد من الهجمات من قبل المهاجمين، مثل هجوم الثقب الأسود Blackhole attack[19].

قد لا يكون الاتصال ما بين المتحكم والمبدل مباشراً وإنما عن طريق مجموعة من أجهزة التبديل الأُخرى. لذلك فإن كل المبدلات والمضيفين المتصلة معهم في مسار الاتصال قد تصبح عقد مستخدمة من قبل هجوم MIM.

4. سيناريو الهجوم

تمت دراسة شبكة مكونة من ثمانية مضيفين، تمثل العقدتان h1 و h8 العقد الضحية والتي سيطبق عليها هجوم MIM، وتمثل العقدة h2 العقدة الخبيثة التي ستنفذ هذا الهجوم. طبق هجوم الرجل في المنتصف باستخدام الأداة DSniff من أجل السيطرة على جداول ARP الخاصة بالعقد التي يراد مهاجمتها.

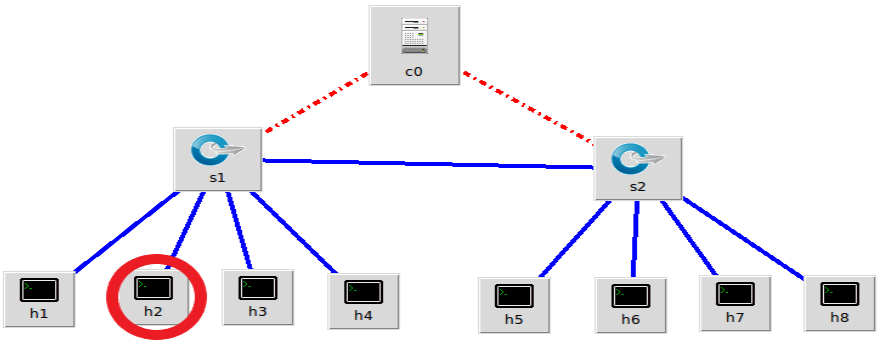
اعتمدت معظم الأبحاث على مهاجمة المتحكم من عقدة من خارج الشبكة (هجوم خارجي) وذلك لأنه عند السيطرة على المتحكم فإنه عندها يمكن السيطرة على كامل الاتصالات ضمن الشبكة. درسنا في هذا البحث حالتين: في الحالة الأولى دُرسَت ثلاثة سيناريوهات اعتمدت على أن إحدى عقد الشبكة تمت السيطرة عليها (هجوم داخلي)،وبعد ذلك قامت العقدة المهاجمة بالعمل وفق سيناريوهات مختلفة بحسب كل طوبولوجيا من الطوبولوجيات المدروسة. أما في الحالة الثانية دُرس الهجوم على المتحكم (هجوم خارجي) وحُسب كل من الإنتاجية والتأخير باستخدام الأدة Cbench.

5. نموذج الشبكة

تتألف الشبكة من المتحكم Floodlight مع توزيع للمبدلات والمضيفين بحسب نوع الطوبولوجيا المدروسة، حيث ستُدرس ثلاثة أنواع من الطوبولوجيات في شبكات SDN هي الطوبولوجيا الشجرية والطوبولوجيا ذات المبدل الوحيد والطوبولوجيا الخطية. سيطبق في السيناريوهات الثلاثة السابقة الذكر الهجوم بين المضيفين في الشبكة، أما في السيناريو الرابع سيدرس التأخير والإنتاجية الخاصة بالمتحكم في حال تعرضه للهجوم بشكل مباشر.

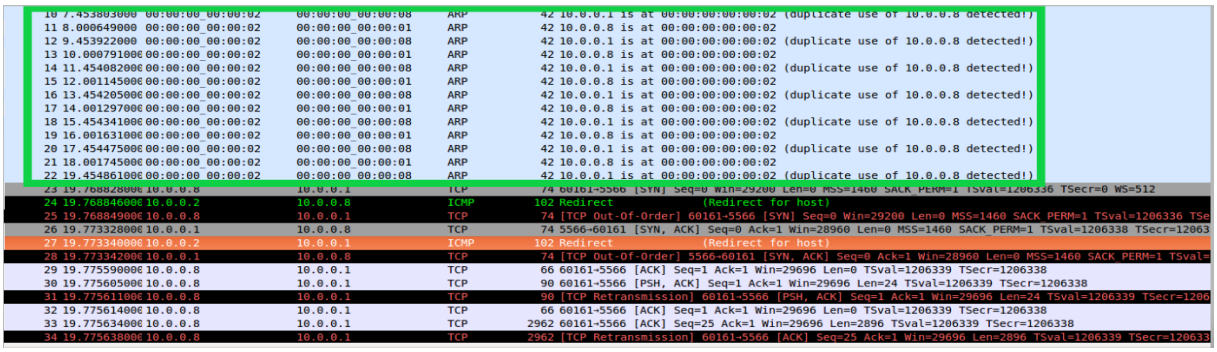
1.5 السيناريو الأول

سندرس في هذا السيناريو الطوبولوجيا الشجرية Treetopology المبينة في الشكل (7) سنعتمد على وجود متحكم مع مبدلين و ثمانية مضيفين كل أربعة منها متصلة مع مبدل، وعرض حزمة الوصلات بين عناصر الشبكة هو 200Mbit، وبفرض أن الوصلات دون خسارة أو تأخير.



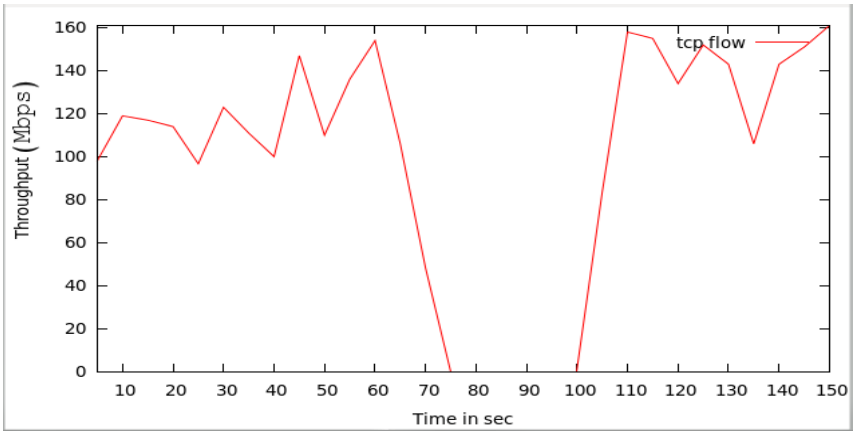
الشكل(7): شبكة SDN ذات طوبولوجيا شجرية

في المرحلة الأولى تقوم العقدة المهاجمة h2 بالتنصت على الاتصالات ما بين العقدتين المهاجمتَين. من خلال الواجهة التالية باستخدام برنامج Wireshark نلاحظ كيفية مرور جميع الرزم المرسلة من العقدة المصدر ذات العنوان 10.0.0.1 إلى العقدة الهدف ذات العنوان 10.0.0.8 عبر المهاجم ذو العنوان 10.0.0.2حيث تعرض الواجهة أنه في الحالة البدائية (ضمن المستطيل الأخضر) كيفية إرسال رزم ARP بشكل مستمر من أجل السيطرة على الاتصال وعندما تبدأ إحدى العقدتين بالإرسال نلاحظ كيفية بدء توجيه هذه الرزم عن طريق العقدة المهاجمة h2 كما هو مبين في الشكل (8).



الشكل (8): تنفيذ هجوم الرجل في المنتصف

من ثم تم تطوير الهجوم من خلال منع الخدمة بشكل نهائي عن طريق إنشاء اتصال بين المضيفين لمدة ثلاث دقائق بحيث يكون الاتصال في الدقيقة الأولى دون وجود هجوم، ثم يهاجم الاتصال لمدة نصف دقيقة ومن ثم تتم إعادة الاتصال فيما تبقى من زمن الاتصال. عند رسم مخطط الإنتاجية نحصل على الشكل(9)



الشكل (9): مخطط الانتاجية لشبكة شجرية قبل وأثناء وبعد تنفيذ هجوم حجب الخدمة

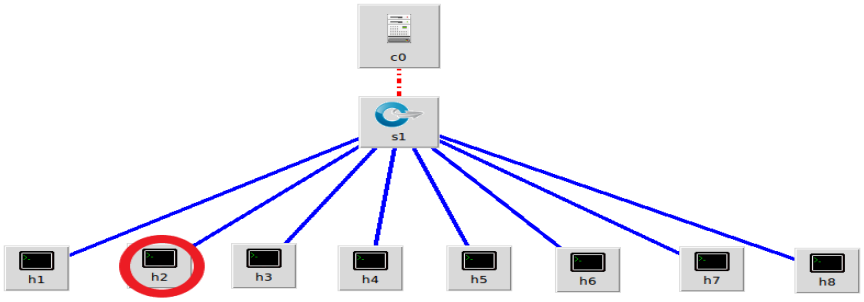
نلاحظ من خلال المخطط أنه لمدة 60 ثانية تقريباً كان يتم الاتصال بشكل طبيعي بين العقدتين h1 و h8،وعند بدء الهجوم في الثانية 60 نلاحظ كيفية انخفاض الإنتاجية حتى تصل إلى قيمة الصفر في الثانية 75 تقريباً وذلك بسبب هجوم حجب الخدمة المطبق بين هذين المضيفين، وبعد إيقاف الهجوم في الثانية 90 نلاحظ أن الشبكة احتاجت مدة

زمنية حوالي 10 ثوان حتى تمكنت من استعادة الاتصال ومن ثم بدأت الإنتاجية بالتزايد بشكل مشابه للحالة ما قبل الهجوم.

2.5 السيناريو الثاني

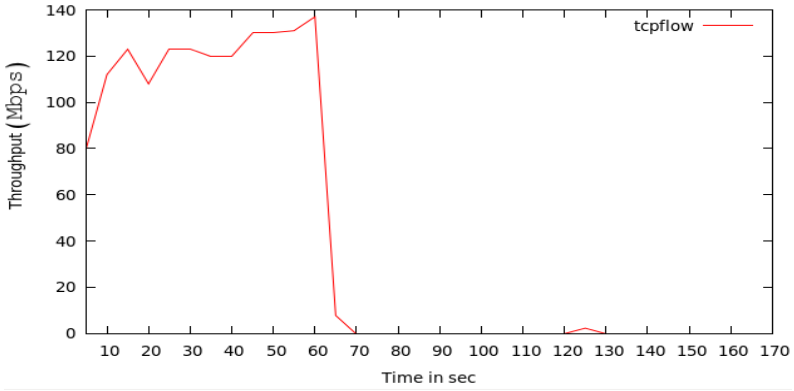
سندرس في هذا السيناريو الطوبولوجيا ذات المبدل الوحيد Single Switch topology المبينة في الشكل (10) سوف نعتمد على وجود متحكم مع مبدل وحيد وثمانية مضيفين متصلة معه، وعرض حزمة الوصلات بين عناصر الشبكة هو 200Mbit، وبفرض أن الوصلات دون خسارة أو تأخير.

اعتمد في هذه الطوبولوجيا على سيناريو مختلف عن الشبكة السابقة حيث أنشأنا اتصال بين المضيفين لمدة ثلاث دقائق بحيث يكون الاتصال للدقيقة الأولى دون وجود هجوم ثم يُهاجم الاتصال لمدة دقيقة أخرى ومن ثم يعاد الاتصال فيما تبقى من زمن الاتصال.



الشكل (10): شبكة SDN ذات طوبولوجيا بمبدل وحيد

وعند رسم مخطط الإنتاجية نحصل على الشكل(11)

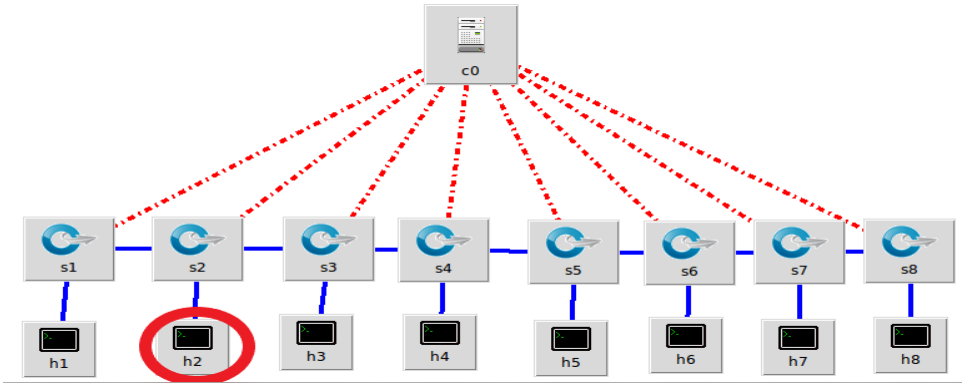


الشكل (11): مخطط الانتاجية لشبكة ذات مبدل وحيد قبل وأثناء وبعد تنفيذ هجوم حجب الخدمة

نلاحظ من خلال المخطط أنه لمدة دقيقة تقريباً كان يتم الاتصال بشكل طبيعي بين العقدتين h1 و h8وعند بدء الهجوم في الثانية 60 نلاحظ انخفاض الإنتاجية حتى تصل إلى قيمة الصفر وذلك بسبب هجوم حجب الخدمة المطبق بين هذين المضيفين وبعد إيقاف الهجوم في اللحظة 120 نلاحظ أن الشبكة لم تتمكن من استعادة الاتصال بين هاتين العقدتين مما أدى إلى حجب الخدمة حتى بعد انتهاء الهجوم.

3.5 السيناريو الثالث

سندرس في هذا السيناريو الطوبولوجيا الخطية Linear topology المبينة في الشكل (12) سوف نعتمد على وجود متحكم متصل مع ثمانية مبدلات و ثمانية مضيفين متصلة معها وعرض حزمة الوصلات بين عناصر الشبكة هو 200Mbit وبفرض أن الوصلات دون خسارة أو تأخير.



الشكل(12): شبكة SDN ذات طوبولوجيا خطية

في هذا السيناريو اعتمدنا على برنامج Wire shark لتحليل الرزم التي ترد إلى المتحكم باستخدام البروتوكول Open flow حيث تمت مراقبة الرزم التي ترد قبل تنفيذ الهجوم كما هو مبين في الشكل (13)

وأثناء تنفيذ الهجوم كما هو مبين في الشكل (14) حيث تم تنفيذ الهجوم باستخدام العقدة h2 على الاتصال ما بين العقدتين h1 و h8 حيث تم إنشاء الاتصال باستخدام الأداة Iperf وذلك باستخدام بروتووكول النقل TCP ولمدة 3 دقائق وكانت النتائج كما يلي:

```
7168 204.3626950€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7169 204.3627170€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7170 204.3627420€ 127.0.0.1  127.0.0.1  TCP        66 6653→35293 [ACK] Seq=45376 Ack=27543 Win=3327 Len=0 TSval=44005 TSecr=44005
7171 204.3632730€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7172 204.3633300€ 127.0.0.1  127.0.0.1  TCP        66 6653→35289 [ACK] Seq=43609 Ack=20215 Win=4734 Len=0 TSval=44005 TSecr=44005
7173 204.3636290€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7174 204.3639560€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7175 204.3639820€ 127.0.0.1  127.0.0.1  TCP        66 6653→35286 [ACK] Seq=45731 Ack=33963 Win=4222 Len=0 TSval=44006 TSecr=44006
7176 204.3639870€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7177 204.3643480€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7178 204.3645330€ 127.0.0.1  127.0.0.1  TCP        66 6653→35294 [ACK] Seq=214920 Ack=197740 Win=4862 Len=0 TSval=44006 TSecr=44006
7179 204.9593590€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7180 204.9597210€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7181 204.9597610€ 127.0.0.1  127.0.0.1  TCP        66 6653→35291 [ACK] Seq=24602 Ack=26925 Win=4222 Len=0 TSval=44155 TSecr=44155
7182 205.1828270€ 127.0.0.1  127.0.0.1  OpenFlow  167 Type: OFPT_PACKET_OUT
7183 205.1829580€ 127.0.0.1  127.0.0.1  OpenFlow  167 Type: OFPT_PACKET_OUT
7184 205.1830420€ 127.0.0.1  127.0.0.1  OpenFlow  167 Type: OFPT_PACKET_OUT
7185 205.1831260€ 127.0.0.1  127.0.0.1  OpenFlow  167 Type: OFPT_PACKET_OUT
7186 205.1831850€ 127.0.0.1  127.0.0.1  OpenFlow  167 Type: OFPT_PACKET_OUT
7187 205.1841520€ 127.0.0.1  127.0.0.1  TCP        66 35290→6653 [ACK] Seq=33714 Ack=45243 Win=1216 Len=0 TSval=44211 TSecr=44210
7188 205.1845090€ 127.0.0.1  127.0.0.1  OpenFlow  169 Type: OFPT_PACKET_IN
7189 205.1845380€ 127.0.0.1  127.0.0.1  TCP        66 6653→35286 [ACK] Seq=45933 Ack=34066 Win=4222 Len=0 TSval=44211 TSecr=44211
7190 205.1849990€ 127.0.0.1  127.0.0.1  OpenFlow  169 Type: OFPT_PACKET_IN
7191 205.1850240€ 127.0.0.1  127.0.0.1  TCP        66 6653→35288 [ACK] Seq=44823 Ack=33758 Win=4222 Len=0 TSval=44211 TSecr=44211
7192 205.1852700€ 127.0.0.1  127.0.0.1  OpenFlow  169 Type: OFPT_PACKET_IN
7193 205.1852970€ 127.0.0.1  127.0.0.1  TCP        66 6653→35291 [ACK] Seq=24602 Ack=27028 Win=4222 Len=0 TSval=44211 TSecr=44211
```

الشكل (13): الرزم الواردة إلى المتحكم في حال عدم وجود هجوم

```
7333 211.7584880€ 127.0.0.1  127.0.0.1  TCP        66 6653→35291 [ACK] Seq=24937 Ack=27052 Win=4222 Len=0 TSval=45854 TSecr=45854
7334 213.6599390€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7335 213.6599450€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7336 213.6603930€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7337 213.6607540€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7338 213.6608080€ 127.0.0.1  127.0.0.1  TCP        66 6653→35290 [ACK] Seq=45384 Ack=33952 Win=3327 Len=0 TSval=46330 TSecr=46330
7339 213.6618430€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7340 213.6618940€ 127.0.0.1  127.0.0.1  TCP        66 6653→35286 [ACK] Seq=46175 Ack=34201 Win=4222 Len=0 TSval=46330 TSecr=46330
7341 213.6620100€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7342 213.6620350€ 127.0.0.1  127.0.0.1  TCP        66 6653→35293 [ACK] Seq=45820 Ack=27781 Win=3327 Len=0 TSval=46330 TSecr=46330
7343 213.7588090€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7344 213.7590970€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7345 213.7593930€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7346 213.7595650€ 127.0.0.1  127.0.0.1  TCP        66 6653→35294 [ACK] Seq=215263 Ack=197875 Win=4862 Len=0 TSval=46355 TSecr=463
7347 213.7596390€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7348 213.7596900€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7349 213.7597160€ 127.0.0.1  127.0.0.1  TCP        66 6653→35289 [ACK] Seq=44053 Ack=20453 Win=4734 Len=0 TSval=46355 TSecr=46355
7350 213.7597690€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7351 213.7597860€ 127.0.0.1  127.0.0.1  TCP        66 6653→35288 [ACK] Seq=45267 Ack=33893 Win=4222 Len=0 TSval=46355 TSecr=46355
7352 213.7600710€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7353 213.7603030€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7354 213.7603420€ 127.0.0.1  127.0.0.1  TCP        66 6653→35292 [ACK] Seq=59845 Ack=42547 Win=3327 Len=0 TSval=46355 TSecr=46355
7355 213.8588170€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REQUEST
7356 213.8597680€ 127.0.0.1  127.0.0.1  OpenFlow   74 Type: OFPT_ECHO_REPLY
7357 213.8598500€ 127.0.0.1  127.0.0.1  TCP        66 6653→35291 [ACK] Seq=24945 Ack=27060 Win=4222 Len=0 TSval=46380 TSecr=46380
```

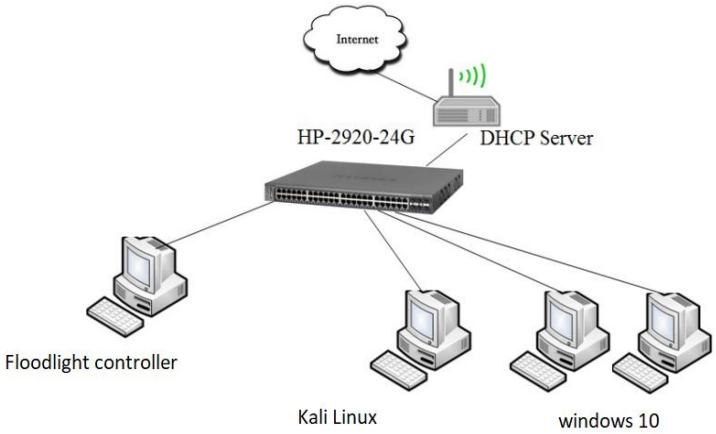
الشكل (14): الرزم الواردة إلى المتحكم في حال وجود هجوم

من الحالة الأولى الممثلة في الشكل (13) نلاحظ أنه في حال عدم وجود هجوم فإنه يتم تبادل 4 أنواع من الرزم: الرزمتان OFPT_ECHO_REQUEST و OFPT_ECHO_REPLY واللتان يتم تبادلهما بكلا الاتجاهين بين المبدلات والمتحكم لمعرفة مدى صلاحية الاتصال، والرزمتان OFPT_PACKET_IN و OFPT_PACKET_OUT من أجل الدلالة على إرسال الرزمة إلى المتحكم عند عدم مطابقتها للقواعد الموجودة في جداول التدفق.

أما في الشكل (14) أي بعد تنفيذ هجوم حجب الخدمة فإننا نلاحظ أنه فقط يتم تبادل رزمتين OFPT_ECHO_REQUEST و OFPT_ECHO_REPLY ولا يوجد أي تبادل للرزم OFPT_PACKET_IN وOFPT_PACKET_OUT وذلك يدل على أنه لا يوجد اتصال في الشبكة لأنه لا يتم إرسال أية رسائل أخرى إلى المتحكم.

3.5 السيناريو الرابع

تمت دراسة الشبكة المبينة في الشكل (15) والتي تتكون من مخدم DHCP (Dynamic Host Configuration Protocol)[20]الذي يقوم بتوزيع عناوين IP بشكل ديناميكي وتم الاتصال بهذه الشبكة من قبل نظام تشغيل خاص بالمتحكم Floodlight و نظام KaliLinux يمثل العقدة المهاجمة وأجهزة تعمل بنظام Windows.



الشكل (15): نموذج الشبكة المستخدم في السيناريو الرابع

نفذ هجوم الرجل في المنتصف على المتحكم من قبل مهاجم يستخدم نظام Kalilinux باستخدام الأداة Ettercapعلى الوصلة مابين المتحكم وبوابة العبور gateway ودُرست الإنتاجية والتأخير باستخدام الأداة Cbench قبل وبعد تنفيذ الهجوم حيث تم تشغيل المتحكم بفرض وجود 8 مبدلات وتكرار العملية مرتين وحصلنا على النتائج الآتية

- قبل تنفيذ الهجوم

التأخير

```
floodlight@floodlight:~$ cbench -c localhost  -s 4 -l 2 -p 6653
cbench: controller benchmarking tool
   running in mode 'latency'
   connecting to controller at localhost:6653
   faking 4 switches offset 1 :: 2 tests each; 1000 ms per test
   with 100000 unique source MACs per switch
   learning destination mac addresses before the test
   starting test with 0 ms delay after features_reply
   ignoring first 1 "warmup" and last 0 "cooldown" loops
   connection delay of 0ms per 1 switch(es)
   debugging info is off
15:22:43.181 4   switches: flows/sec:  82  85  78  72   total = 0.317000 per ms
15:22:44.298 4   switches: flows/sec:  99  101  96  88   total = 0.380708 per ms
RESULT: 4 switches 1 tests min/max/avg/stdev = 380.71/380.71/380.71/0.00 responses/s
floodlight@floodlight:~$
```

الإنتاجية

```
floodlight@floodlight:~$ cbench -c localhost  -s 4 -l 2 -t -p 6653
cbench: controller benchmarking tool
   running in mode 'throughput'
   connecting to controller at localhost:6653
   faking 4 switches offset 1 :: 2 tests each; 1000 ms per test
   with 100000 unique source MACs per switch
   learning destination mac addresses before the test
   starting test with 0 ms delay after features_reply
   ignoring first 1 "warmup" and last 0 "cooldown" loops
   connection delay of 0ms per 1 switch(es)
   debugging info is off
15:22:09.200 4   switches: flows/sec:  119  124  125  160   total = 0.527626 per ms
15:22:10.303 4   switches: flows/sec:  131  120  113  118   total = 0.480811 per ms
RESULT: 4 switches 1 tests min/max/avg/stdev = 480.81/480.81/480.81/0.00 responses/s
floodlight@floodlight:~$
```

- بعد تنفيذ الهجوم

التأخير

```
floodlight@floodlight:~$ cbench -c localhost  -s 4 -l 2 -p 6653
cbench: controller benchmarking tool
   running in mode 'latency'
   connecting to controller at localhost:6653
   faking 4 switches offset 1 :: 2 tests each; 1000 ms per test
   with 100000 unique source MACs per switch
   learning destination mac addresses before the test
   starting test with 0 ms delay after features_reply
   ignoring first 1 "warmup" and last 0 "cooldown" loops
   connection delay of 0ms per 1 switch(es)
   debugging info is off
15:23:57.527 4   switches: flows/sec:  153  166  153  130   total = 0.600354 per ms
15:23:58.632 4   switches: flows/sec:  149  162  162  152   total = 0.621928 per ms
RESULT: 4 switches 1 tests min/max/avg/stdev = 621.93/621.93/621.93/0.00 responses/s
```

الإنتاجية

```
floodlight@floodlight:~$ cbench -c localhost  -s 4 -l 2 -t -p 6653
cbench: controller benchmarking tool
   running in mode 'throughput'
   connecting to controller at localhost:6653
   faking 4 switches offset 1 :: 2 tests each; 1000 ms per test
   with 100000 unique source MACs per switch
   learning destination mac addresses before the test
   starting test with 0 ms delay after features_reply
   ignoring first 1 "warmup" and last 0 "cooldown" loops
   connection delay of 0ms per 1 switch(es)
   debugging info is off
15:25:34.958 4   switches: flows/sec:  100  116  87  98   total = 0.399801 per ms
15:25:36.060 4   switches: flows/sec:  101  100  94  99   total = 0.393353 per ms
RESULT: 4 switches 1 tests min/max/avg/stdev = 393.35/393.35/393.35/0.00 responses/s
```

من النتائج السابقة نلاحظ أن القيمة المتوسطة للتأخير قبل الهجوم هي 380.71 استجابة/ثا بينما بعد تنفيذ الهجوم هي 621.93 استجابة/ثانية أي يوجد تأخير أكبر مقارنة بالحالة ما قبل الهجوم إذ ازداد التأخير بنسبة 63.36% أما بالنسبة للإنتاجية نلاحظ أنه قبل الهجوم القيمة المتوسطة هي 480.81 بينما بعد الهجوم 393.35 أي يوجد انخفاض في الإنتاجية بنسبة 1.82%.

الاستنتاجات والتوصيات:

بعد هذه الدراسة، يمكننا التوصل للاستنتاجات الآتية:

- تمت دراسة ثلاثة سيناريوهات في طوبولوجيات مختلفة، وبينت النتائج في السيناريو الأول أنه في حال كانت مدة الهجوم صغيرة بالنسبة للاتصال المبني فإن الشبكة ستكون قادرة على إعادة الاتصال بين المضيفين، أما في السيناريو الثاني فبينت النتائج أنه في حال كانت مدة الهجوم طويلة مقارنة بمدة الاتصال فإن الشبكة لن تكون قادرة على إعادة الاتصال كما كان قبل الهجوم، وفي السيناريو الثالث ومن خلال مراقبة رزم البروتوكول Open Flow في حال وجود هجوم لاحظنا أن العقد لا ترسل أي رزم إطلاقاً إلى المتحكم كدلالة على خروجها عن الخدمة.
- أما في السيناريو الرابع الذي قمنا فيه بمهاجمة المتحكم بشكل مباشر فقد وجدنا أن الهجوم يؤدي إلى خفض الإنتاجية وزيادة في التأخير ولكن المهاجم لم يكن قادراً على حجب الخدمة بشكل نهائي وذلك لأن المتحكم يتضمن وحدات خاصة قابلة للتعامل مع الهجمات الخارجية.

يمكن تلخيص التوصيات في النقاط الآتية:

- العمل على دراسة تطوير للبروتوكول ARP بحيث يمكن مواجهة الهجمات المعتمدة على خداع ARPفي حال طبقت سواءً على الوصلات ما بين العقد أو على الوصلة ما بين المتحكم والمبدل مثل هجمات الرجل في المنتصف وحجب الخدمة.
- العمل على جعل هذا التطوير قادراً على اكتشاف الهجوم خلال زمن منخفض من أجل التعامل مع المهاجم والحد من تأثيره السلبي على الشبكة.

References:

[1] D. Rana and S. Chamoli, "Software Defined Networking (SDN) Challenges, issues and Solution", in International Journal of Computer Sciences and Engineering, February 2019.

[2] J. Meghana, T.Subashri and K.Vimal ; "A Survey on ARP Cache Poisoning And techniques for detection and mitigation", inInternational Conference on Signal Processing, Communications and Networking (ICSCN-2017), India, March 2017.

[3] W.Xia, Y.Wen, C.Foh, D.Niyato and H.Xie, "a survey on software-defined networking", IEEE communication surveys & tutorials, vol. 17, no. 1, first quarter 2015.

[4] floodlight.atlassian.net.availableat: https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/pages/40403023/web+gui. Last visit 1 August 2019

[5] http://www.projectfloodlight.org/floodlight/.Last visit 18 September 2019.

[6] https://iperf.fr/iperf-download.php. Last visit 7 October 2019.

[7] https://sectools.org/tool/dsniff/ . Last visit 7 October 2019.

[8] CBench, available at: https://github.com/mininet/oflops/tree/master/cbench.. Last Visit 1 October 2019.
[9] A.M.Abdelsalam, A.El-Sisi and V.Reddy, "Mitigating ARP Spoofing Attacks in Software-Defined Networks", in ICCTA ,at Alexandria,Egypt,2015.
[10] T. Alharbi, D. Durando, F.Pakzad and M.Portmann, "Securing ARP in Software Defined Networks", in IEEE 41st conference on local computernetworks, 2016.
[11] L. Ertaul and K. Venkatachalam, "Security of Software Defined Networks (SDN)", in int'l conf. wireless networks |icwn'17, isbn: 1-60132-462-6, 2017.
[12] Open Networking Foundation (ONF). [online]. available: https://www.opennetworking.org/. Last visit 15 oct 2019.
[13] M .Sabih, A .Abu Obaid, Evaluating the Performance of Controllers in Software-Defined Networks, Tishreen University Journal for Research and Scientific Studies, Fifth Issue, 2018.
[14] https://whatis.techtarget.com/definition/link-layer-discovery-protocol-LLDP. Last visit, 25 October 2019.
[15] S.Sharma, W.Tavernier, S. sahhaf and P.Demeester. "Verification of Flow Matching Functionality in the Forwarding Plane of OpenFlow Networks", in IEICE TRANS. COMMUN., Vol.E98–B, NO.11 November 2015.
[16] S.Asadollahi and DrB.Goswami, "Experimenting with scalability of floodlight controller in software definednetworks", International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), 2017.
[17] Di Li, A. Vasilakos and W. Jiafu, "Security in Software-Defined Networking:Threats and Countermeasures," Mobile Networks and Applications, January 2016.
[18] I. Paul, "Lenovo preinstalls man-in-the-middle adware that hijacks HTTPS traffic on new PCs". PC World, 19 February 2015.
[19] A.Khalil, R.Badlishah, N. Yaakob, M. Hafiz and M.Elshaikh. "Black hole attack behavioral analysis general network scalability", in Indonesian Journal of Electrical Engineering and Computer Science, Vol. 13, No. 2 February 2019.
[20] https://docs.microsoft.com/en-us/windows-server/networking/technologies/dhcp/dhcp-top. Last visit, 25 October 2019.