14. Spring Boot security : site. — URL: https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/#web.security (дата обращения: 21.04.2022). — Текст : электронный.
15. Provos N. A Future-Adaptable Password Scheme / N. Provos, D. Mazières. — Text : direct // Proceedings of 1999 USENIX Annual Technical Conference. — 1999. — P. 81-92.
16. Malvoni K. Are your passwords safe: energy-efficient bcrypt cracking with low-cost parallel hardware / K. Malvoni, S. Designer, J. Knezovic. — Text : direct // Proceedings of the 8th USENIX Conference on Offensive Technologies. — 2014. — P. 6.

***А. И. Угренинов, А. М. Шабалин***
*Тюменский государственный университет, г. Тюмень*

**УДК 004.732**

## АВТОМАТИЗАЦИЯ СЕТЕВЫХ СЕРВИСОВ ПРЕДПРИЯТИЯ СРЕДСТВАМИ SDN

***Аннотация.*** *В работе описан анализ различных протоколов, предназначенных для централизованного управления сетью. Представляются варианты использования автоматизации для решения различных задач по мониторингу, администрированию и обеспечению безопасности сети. Результатом работы стала модель компьютерной сети, решающая задачу проверки доступности сетевой среды средствами автоматизации.*

***Ключевые слова:*** *автоматизация, программно-управляемые сети, SDN, Netconf.*

**Введение.** В связи с ежегодным ростом использования сетей возрастает и потребность в увеличении количества устройств, а с ними появляется и большое количество проблем, связанных с управлением и отслеживанием всех ее элементов [1]. Увеличиваются затраты на администрирование, а также шансы появления ошибок конфигурации, которые могут привести к проблемам безопасности и нестабильности сети в целом. Одним из решений этих проблем является внедрение автоматизации, которая используется для исключения выполнения однотипных операций и используемая в качестве инструмента автоматической конфигурации устройств, что позволяет избежать одной из самых частых причин — ошибок

в конфигурации, а именно человеческого фактора, который может составлять от 40% всех ошибок в сети [2]. Тем же образом она используется для создания архитектур автоматизированного развертывания с применением таких программных средств, как Ansible [3].

**Проблема исследования.** Для уменьшения влияния человеческого фактора и усиления защиты было решено автоматизировать и упростить задачи, связанные с работой сети. Для решения этих вопросов были выделены следующие задачи:

1. Рассмотреть концепцию программно-управляемых сетей.
2. Изучить вопрос возможности унифицированной конфигурации устройств, которая не будет иметь зависимости от производителей.
3. Сравнить методы конфигурации и мониторинга устройств.
4. Реализовать функции защиты сети.

**Материалы и методы.** Методами данного исследования стали: изучение, сравнение, эксперимент. Для развертывания сети был выбран эмулятор EVE-NG, а в качестве устройств взяты образы cisco CSR1000v ver. 17.01.01. В качестве контроллера выступал виртуальный хост с наличием python и библиотекой ncclient.

**Результат.** Программно-управляемые сети (Software Defined Network, SDN) — это концепция, в которой происходит разделение уровня управления от уровня передачи данных, что позволяет управлять устройствами централизованно (рис. 1).

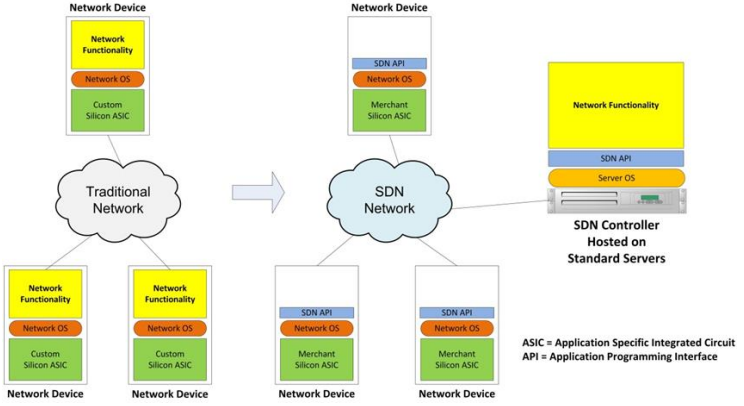


*Рис. 1.* Архитектура сети SDN

Центром такой сети является один или группа контроллеров, которые с помощью специальных интерфейсов и протоколов взаимодействуют с сетевыми устройствами, а с помощью API позволяют аккумулировать данные со всей сети, предоставляя возможность для удобного управления и автоматизации [4].

Самыми частыми и используемыми интерфейсами взаимодействия с устройствами являются протоколы SNMP, Netconf и CLI, сравнение которых приведено на рис. 2.

| | SNMP | CLI | NETCONF |
|---|---|---|---|
| **Модель данных** | MIB | - | Модули |
| **Язык моделей данных** | SMI | - | YANG |
| **Способ управления конфигурацией** | SNMP SET/GET запросы | Ручной ввод или написание скриптов | NETCONF |
| **RPC (Remote Procedure Call)** | BER | Пальцы | XML |
| **Транспорт** | UDP | Telnet, SSH, Serial... | SSH, TLS |

*Рис. 2*. Сравнение интерфейсов взаимодействия

В контексте использования интерфейса для автоматизации интерфейс командной строки смотрится не слишком универсальным. Для взаимодействия с ним необходимо обладать знанием синтаксиса, который может меняться в зависимости от версии и вендора устройства.

Протокол SNMP на данный момент поддерживается практически всеми устройствами, но имеет некоторые ограничения из-за своей модели данных MIB. Из-за того, что MIB является старой моделью, она разрослась настолько, что использование ее является не самой тривиальной задачей. Также в SNMP отсутствует стандарт автоматической проверки используемых устройством MIB, а некоторый новый функционал все еще не поддерживается [5].

Netconf является самым молодым из представленных интерфейсов. Netconf, как и SNMP поддерживается очень большим количеством устройств. Он несет в своей основе множество стандартизированных модулей, а для более тонкой настройки и проприетарных

протоколов также существуют и модули от производителей. Netconf имеет стандарт проверки поддерживаемых устройством модулей. Все вышеперечисленное делает его хорошим кандидатом для использования в целях автоматизации.

Было решено реализовать следующие практические задачи, решающие проблемы доступности в компьютерной сети: отслеживание состояния интерфейсов, автоматизированная конфигурация устройств с помощью протокола Netconf, создание визуализации топологии сети, а также отслеживание неавторизованных mac-адресов.

Для использования протокола была выбрана Python библиотека ncclient, которая позволяет формировать запросы к устройствам через Netconf.

Предварительная настройка устройств включает в себе несколько пунктов: включение функций Netconf на устройстве, создание пользователя для входа, включение функций ssh и предоставлении возможности подключения к устройству удаленно (рис. 3).

```
R1#show run | section netconf
username netconfadmin privilege 15 secret 9 $9$yKXG4DIwkNymuE$d.d85TblzFDsiREbUN
sWy1x5vpWSBSh8dS930AMR8yc
netconf-yang
R1#
R1#sh run | sec ssh
ip ssh rsa keypair-name ssh-key
ip ssh version 2
 transport input ssh
R1#
```

*Рис. 3.* Пример конфигурации роутера CSR1000v

Поочередно через сеть скрипт подключается к каждому устройству и с помощью операции get получает информацию о состоянии интерфейсов, включая информацию о соседях, полученных с помощью протокола CDP. С помощью все той же операции, в зависимости от типа устройств запрашивает информацию об известных mac-адресах и интерфейсах, с которых были изучены эти адреса.

На основе информации, полученной ранее, строится и выводится топология сети, а для каждого устройства отображается информация о состоянии интерфейсов и о наличии подключенных на них неавторизованных устройств для их локализации в сети и дальнейшем

их изучении (рис. 4). Таким образом можно следить за состоянием сети и предупреждать атаки, связанные с изменением топологии, или детектировать вторжения.

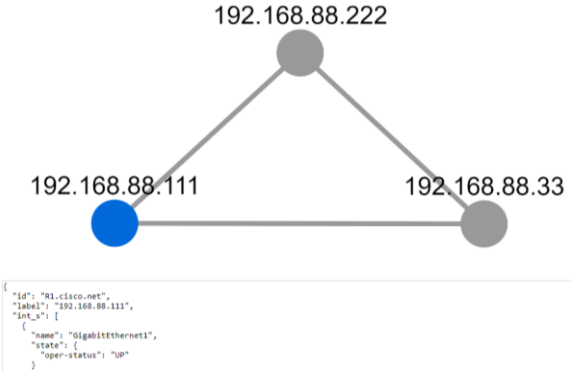


*Рис. 4.* Топология сети, построенная на основе CDP

**Заключение.** В результате проведенной работы была рассмотрена концепция программно-управляемых сетей. В статье дано сравнение протоколов конфигурации, продемонстрированы некоторые возможности автоматизации сервисов, которые помогают уменьшить количество возможных путей атак на сеть, а также достигнута гибкость в управлении и мониторинге.

**СПИСОК ЛИТЕРАТУРЫ**

1. Fixed network data traffic: worldwide trends and forecasts 2020-2026 / R. Wood, J. Konieczny. — URL: https://www.analysysmason.com/research/ content/regional-forecasts-/fixed-network-data-rdfi0-rdmb0/#:~:text=Fixed%20data%20traffic%20grew%20by,half%20of%20all%20countries%20modelled (date of the application: 06.08.2021). — Text : electronic.
2. Top 7 reasons for network automation/ T. Slattery. — URL : https://netcraftsmen.com/top-7-reasons-for-network-automation/ (date of the application: 08.05.2010). — Text : electronic.
3. Automated Provisioning of Application in IAAS Cloud using Ansible Configuration Management / K. Nishat, T. Sanjeev, C. Himanshu, N. Himanshu. — Text : direct // 1st International Conference on Next Generation Computing Technologies. — Dehradun, 2015.

4. Hybrid SDN Networks: A Survey of Existing Approaches / R. Amin, M. Reisslein, N. Shah. — Text : direct // IEEE Communications Surveys & Tutorials. — Vol. 20, Issue 4. — Fourthquarter, 2018.
5. Why use NETCONF/YANG when you can use SNMP and CLI? / Christos Rizos. — URL: https://www.snmpcenter.com/why-use-netconf/ (date of the application: 25.10.2016). — Text : electronic.

***И. М. Пономарев, А. А. Селиверстов, Г. А. Пеннер, И. Р. Зулькарнеев***
*Тюменский государственный университет, г. Тюмень*

**УДК 004.056**

# ПРОЕКТИРОВАНИЕ КИБЕРПОЛИГОНА В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

***Аннотация.*** *В статье представлена концепция обучающего киберполигона в области информационной безопасности. Авторами проанализирован рынок существующих киберполигонов и сделан вывод о необходимости создания решения с возможностью обучения. Описаны требования, предъявляемые к проектируемому киберполигону, и его режимы работы. Разработана математическая модель рейтинговой системы для соревновательного режима. В завершении обозначены дальнейшие направления развития.*

***Ключевые слова:*** *информационная безопасность, обучение, киберполигон, red team, blue team.*

**Введение.** По данным исследования Всероссийского научно-исследовательского института труда, проведенного в 2019 г., в Российской Федерации потребность в специалистах по информационной безопасности удовлетворена лишь наполовину, что говорит об их дефиците на рынке труда [1]. При этом, согласно информации «Лаборатории Касперского», количество инцидентов информационной безопасности в российских компаниях увеличилось в 4 раза за первые три месяца 2022 г. по отношению к показателю аналогичного периода 2021 г. [2]. Также Национальным координационным центром по компьютерным инцидентам 24 февраля уровень киберугрозы для России был оценен как «Критический» [3].