D. S. Chernyavskiy

Information Security Policy Modeling for Network Security Systems

Keywords: information security policy, policy management process, network security system

Policy management for network security systems (NSSs) is one of the most topical issues of network security management. Incorrect configurations of NSSs lead to system outages and appearance of vulnerabilities. Moreover, policy management process is a time-consuming task, which includes significant amount of manual work. These factors reduce efficiency of NSSs' utilization. The paper discusses peculiarities of policy management process and existing approaches to policy modeling, presents a model aimed to formalize policies for NSSs independently on NSSs' platforms and select the most effective NSSs for implementation of the policies.

Д. С. Чернявский

МОДЕЛИРОВАНИЕ ПОЛИТИК ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ДЛЯ СЕТЕВЫХ СРЕДСТВ ЗАЩИТЫ ИНФОРМАЦИИ

Введение

В настоящее время самыми распространенными сетевыми средствами защиты информации (ССЗИ) являются межсетевые экраны (МЭ) и системы обнаружения/предотвращения вторжений (IDS/IPS-системы) [1, 2]. Однако существующие подходы к управлению сетевой информационной безопасностью (ИБ) приводят к значительным рискам, связанным с появлением уязвимостей, и отказам систем вследствие ошибок конфигурирования, а основными причинами снижения эффективности управления ССЗИ являются трудоемкие ручные процессы и недостаток наглядности политик [3]. Например, количество правил в политиках ИБ в МЭ Check Point и Cisco может достигать нескольких тысяч, что является одной из главных причин возникновения ошибок конфигурирования [4, 5]. В то же время одними из главных факторов, учитываемых при выборе ССЗИ, являются их стоимость, простота применения, функциональная совместимость [1].

Таким образом, тенденция к увеличению числа ССЗИ и их функциональных возможностей, с одной стороны, должна повысить эффективность противодействия сетевым угрозам посредством выбора из множества ССЗИ с подходящим набором функциональных возможностей наиболее эффективного из них. Однако, с другой стороны, с увеличением сложности ССЗИ и ростом их числа в организации процессы управления политиками (здесь и далее под термином «политика» понимается политика ИБ) для ССЗИ требуют дополнительных временных и денежных затрат, связанных с большим объемом ручной работы, а также с обучением разработчиков политик языкам и средствам задания политик для каждой конкретной платформы ССЗИ.

Процесс управления политиками

Модели процесса управления политиками ИБ представлены в [6–9]. Данные модели являются итеративными и включают в себя схожие операции. Наиболее детальное описание процесса представлено в [6]. Согласно [6], процесс управления политиками разделяется на следующие этапы:

1) *Оценка*. Данный этап может быть инициирован в случае первой итерации процесса или в случае необходимости изменений, идентифицированных на подэтапе «Анализ тенденций и управления событиями» этапа «Эксплуатация». Целью данного этапа является оценка изменений относительно существующих политик и условий среды. Данный этап состоит из двух подэтапов:

a) Оценка политик. На данном подэтапе осуществляется проверка и пересмотр существующих политик, процедур и стандартов. Подэтап состоит из следующих шагов: анализ среды, обнаружение недостатков и противоречий в политиках, обобщение результатов анализа политик, разработка рекомендаций. Результатом подэтапа является решение о том, принимать ли изменения, и оценка того, как данные изменения повлияют на существующие политики.

b) Оценка рисков. На данном подэтапе определяются активы организации и потенциальные угрозы данным активам. Подэтап состоит из следующих шагов:

• Оценка защищенности. На данном шаге определяются подверженные угрозам ИБ элементы в существующей или предполагаемой среде.

• Оценка бизнес-рисков. На данном шаге определяются наиболее ценные информационные активы, подверженные угрозам ИБ.

• Разработка рекомендаций. На данном шаге определяются возможные варианты ССЗИ, затраты на них (включая затраты на персонал), а также приоритеты для данных вариантов и соотношения затрат и выгод.

• Результирующая оценка и итоговые рекомендации. На данном шаге документируются результаты оценки политик и оценки рисков для того, чтобы прийти к решению о принятии изменений. Если изменения приняты, то процесс для данных изменений переходит к этапу «Планирование». Если изменения отклонены, но при этом обнаружено, что необходимо внести другие изменения в политики, то процесс также переходит к этапу «Планирование». Иначе процесс возвращается на этап «Эксплуатация».

2) *Планирование*. На данном этапе осуществляется подготовка к реализации созданных или обновленных политик, а также определение требований для системы обеспечения ИБ (СОИБ), выполнение которых необходимо для реализации данных политик.

a) Разработка политик. Целью данного подэтапа является разработка стратегии и политики ИБ, которые согласуются со стратегией бизнеса. Подэтап состоит из следующих двух шагов:

• Создание/обновление стратегии ИБ. Стратегия ИБ — это обзор будущего направления бизнеса с учетом средств обеспечения ИБ, необходимых для поддержания этого направления. На данном шаге решаются следующие задачи: определение будущих инициатив бизнеса; определение рисков, связанных с данными инициативами; определение средств обеспечения ИБ; назначение приоритетов инициативам ИБ и документирование стратегии ИБ.

• Создание/обновление политик. На данном шаге осуществляются определение области применения политик, написание черновиков политик и их рецензирование.

b) Определение требований. На данном подэтапе организация анализирует политики для определения требований к новой СОИБ с учетом созданных или обновленных политик. Данный подэтап состоит из следующих шагов:

• Преобразование рекомендаций в требования. Высокоприоритетные рекомендации, разработанные на подэтапе «Оценка рисков», используются на данном шаге для формирования требований к СОИБ, необходимых для реализации изменений.

• Разработка детальных требований. Высокоуровневые требования, полученные на предыдущем шаге, детализируются до такого уровня, при котором возможно осуществить выбор ССЗИ. В том числе должны быть рассмотрены вопросы функциональной совместимости систем и сетей.

• Проверка требований. Требования, разработанные на предыдущих двух шагах, проверяются на соответствие входным данным подэтапа «Определение требований». Каждое требование должно отображаться в конкретный риск, определенный на этапе «Оценка рисков», или в некоторое положение политики. Дополнительно на данном этапе проверяются требования регулирующих органов, если таковые существуют.

3) *Реализация*. На данном этапе осуществляется реализация политик.

a) Определение ССЗИ. На данном подэтапе определяются те ССЗИ, которые соответствуют требованиям политики. Подэтап состоит из следующих шагов:

• Проектирование СОИБ. Требования, определенные на этапе «Планирование» используются для проектирования высокоуровневой структуры СОИБ, включая технические и организационные компоненты.

• Определение типа ССЗИ. Высокоуровневые требования преобразуются в требования для ССЗИ, на основе которых определяется требуемый тип ССЗИ (например, МЭ или IDS-система). Организации могут иметь также дополнительные требования к поставщику ССЗИ (например, рассматривать только текущего поставщика или иметь приоритетный список поставщиков), которые учитываются на данном шаге.

• Оценка ССЗИ. На данном шаге проверяются и оцениваются возможные варианты для каждого типа ССЗИ, соответствующие установленным требованиям.

• Выбор ССЗИ. На данном шаге выбирается ССЗИ, наилучшим образом соответствующее требованиям, и включается в проект СОИБ. Список ССЗИ проверяется на предмет отсутствия дублирующих требований, которым соответствуют различные ССЗИ, а также выявляются возможности взаимозаменяемости ССЗИ в рамках СОИБ.

b) Внедрение ССЗИ. На данном подэтапе внедряются ССЗИ, выбранные на предыдущем шаге. Данный подэтап состоит из следующих шагов:

• Разработка плана внедрения. На данном шаге формируется план по реализации ранее разработанного проекта.

• Построение. На данном шаге разрабатываются процедуры для поддержки выбранных ССЗИ. Эти процедуры необходимы для осуществления управления и мониторинга СОИБ. На данном шаге также разрабатываются средства обучения, включая справочные файлы и руководства.

• Тестирование. После того как СОИБ построена (обновлена), она должна быть протестирована для того, чтобы гарантировать, что проект полностью реализован, идентифицированные угрозы предотвращены, а новые уязвимости не появились.

• Развертывание. После тестирования СОИБ развертывается в продуктивной среде. Развертывание включает настройку и установку ССЗИ, а также запуск процессов и процедур. На шаге развертывания необходимо обеспечить соответствие требованиям, установленным в политиках, и отсутствие новых рисков.

4) *Эксплуатация*. Данный этап осуществляется на ежедневной основе. Целью этапа является мониторинг ССЗИ, введенных в действие, обработка инцидентов в случае их возникновения, а также анализ тенденций в бизнесе и технологиях.

a) Наблюдение за функционированием. Целью данного подэтапа является определение ежедневных мероприятий, необходимых для реализации СОИБ политик. Подэтап состоит из следующих шагов, выполняемых параллельно:

• Администрирование и эксплуатация. Данный этап включает администрирование, установку обновлений безопасности для систем и приложений, наблюдение за событиями ИБ, выявление новых уязвимостей и другие административные функции.

• Коммуникация. Данная деятельность включает информирование для различных групп. Например, рассылку о новых уязвимостях для администраторов.

• Расследование. Расследование включает в себя деятельность, необходимую для анализа инцидентов ИБ, определения их причин и проверки фактов, предоставления рекомендаций для действий.

• Консультирование по вопросам ИБ. Консультирование службой ИБ проектных групп организации при разработке новых или усовершенствовании существующих процессов или систем.

• Обеспечение соответствия. Данный шаг включает мероприятия, необходимые для того, чтобы гарантировать выполнение СОИБ требований политик.

b) Анализ тенденций и управление событиями. Целью данного подэтапа является выявление событий и тенденций (как внутренних, так и внешних по отношению к организации), которые могут указывать на необходимость внесения изменений в политики. Заключительной частью данного подэтапа является передача запроса изменений на этап «Оценка». На этап «Оценка» должны быть переданы только те изменения, которые соответствуют установленным критериям.

Если на подэтапах «Оценка политик», «Оценка рисков» или «Разработка политик» обнаружено, что какие-либо из политик больше не нужны, то они должны быть удалены [8]. Процесс управления политиками ИБ является итеративным, в силу постоянных изменений в технологиях, бизнес-среде и юридических требованиях [9].

Подходы к моделированию политик

Подходы к моделированию политик достаточно широко представлены в современной научной литературе. Некоторые из подходов, применимые для моделирования политик для ССЗИ, представлены в [10–20].

Техника моделирования [10] основана на четырех неделимых функциях (оконечное оборудование, каналообразующее оборудование, функции преобразования и фильтрации) и формальном языке, позволяющем задавать политики для данных функций. Однако данными функциями не могут быть описаны такие возможности ССЗИ, как регистрация событий в сети. Кроме того, данный подход не предоставляет средств для нахождения лучшего варианта ССЗИ.

Язык контроля доступа, основанный на синтаксисе языка XML и модели контроля доступа Organization-Based Access Control (OrBAC), являющейся расширением ролевой модели (Role-Based Access Control, RBAC), представлен в [11] и предназначен для задания политик для МЭ. Одним из недостатков данного подхода является то, что он поддерживает только политики авторизации (англ. authorization policy) и не поддерживает политики на основе обязательств (англ. obligation policies). Кроме того, недостатком данного подхода является то, что в качестве параметров сетевого трафика в нем рассматриваются только IP-адреса, протоколы и порты, что не позволяет задавать более сложные политики.

Еще один метод на основе OrBAC представлен в [12]. Политики OrBAC задаются и транслируются в конфигурации конкретных устройств с помощью данного метода. Метод также поддерживает политики, учитывающие контекст (включая контексты, заданные пользователем). Однако если политика не может быть реализована существующей системой, то методы [11, 12] не предоставляют средств для выбора реализующих политики ССЗИ.

Подход, представленный в [13, 14], дает возможность задавать политики OrBAC и их контексты, а также транслировать их в конфигурации систем. Подход также позволяет выбрать наиболее подходящие ССЗИ с учетом их стоимости. Однако в данном подходе выбор осуществляется на основе понятия «почти эквивалентных» функций ССЗИ, но формального критерия их определения не приводится.

В [15, 16] представлен формальный язык задания политик и описан процесс их трансляции в конфигурации систем. Данный подход позволяет формализовывать политики авторизации, политики на основе обязательств и их контексты. Аналогичный подход представлен в [17], однако в нем рассматриваются только политики авторизации. Недостатком обоих подходов является отсутствие возможности выбора ССЗИ.

Язык контроля доступа, представленный в [18], основан на синтаксисе языка SQL и предназначен для описания различных моделей контроля доступа, таких как RBAC или мандатная модель управления доступом.

В [19] представлена модель, позволяющая формализовать системы контроля доступа и анализировать их свойства. Модель позволяет задавать политики авторизации, выявлять конфликты в политиках и ошибки, приводящие к уязвимостям.

Подход, описанный в [20], предназначен для композиции и декомпозиции политик для ССЗИ, функционирующих на различных уровнях модели ISO/OSI, с целью исключения ошибок конфигурирования и снижения трудоемкости управления ими.

К недостаткам подходов [18–20] относятся отсутствие возможности задания политик на основе обязательств, а также отсутствие средств выбора ССЗИ.

Далее представлена модель, позволяющая формализовать политики независимо от платформы ССЗИ, а также осуществить выбор наиболее эффективного ССЗИ для реализации данных политик.

Модель политики ИБ

Под ССЗИ далее подразумевается программное, аппаратное, программно-аппаратное средство или его компонент, прямо или косвенно используемое для защиты сетевых сервисов (ресурсов) организации и передаваемой по вычислительным сетям информации, функционирование которого основывается на анализе сетевого трафика. Функцией ССЗИ является формирование некоторого выхода на основе политики ИБ посредством анализа сетевого трафика.

Пусть P является множеством всех политик, которые могут быть реализованы ССЗИ. По сути, P состоит из текстовых строк, которые являются командами настройки ССЗИ. Если ССЗИ использует графический пользовательский интерфейс, то соответствующие ему политики также могут быть представлены в виде текстовых строк. Например, политика МЭ Check Point (рис. 1) может быть представлена в следующем виде:

«*N=1 Source=Any Destination=Web-server VPN=Any Service=http Action=accept ...*».

Пусть T – множество сетевого трафика (множество последовательностей битов); O – множество возможных формируемых ССЗИ выходов, элементами которого являются сообщения, передаваемые в другие системы (например, запись в журнал регистрации событий, сигнал другому ССЗИ).

Тогда ССЗИ может быть представлено в виде конечного автомата

$$F = < T \times P, S, T \times O, \delta, f >,$$

где $T \times P$ – входной алфавит, S – множество состояний, $T \times O$ – выходной алфавит, $\delta: T \times P \times S \to S$ – функция переходов; $f: T \times P \times S \to T \times O$ – функция выходов. ССЗИ функционирует в дискретные моменты времени τ (рис. 2). Множество S может включать такие параметры, как время, количество соединений, и другие параметры, необходимые для анализа сетевого трафика с учетом состояния соединения. Пакетные фильтры могут быть представлены как автомат с одним состоянием, то есть $|S| = 1$.

NO.	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
1	* Any	Web-server	Any Traffic	http	accept	None	Corporate-gw	* Any	Allow connections to corporate web server
2	Internal-net	SQL-Server	Any Traffic	MS-SQL	accept	None	Corporate-gw	* Any	Allow connections from internal network to corporate database server
3	Admin-subnet	Management	Any Traffic	CPMI	accept	None	Corporate-gw	* Any	Allow connections from administrator's network to Check Point management server
4	* Any	* Any	Any Traffic	* Any	drop	None	Corporate-gw	* Any	Drop all other traffic

Рис. 1. Пример политики МЭ Check Point

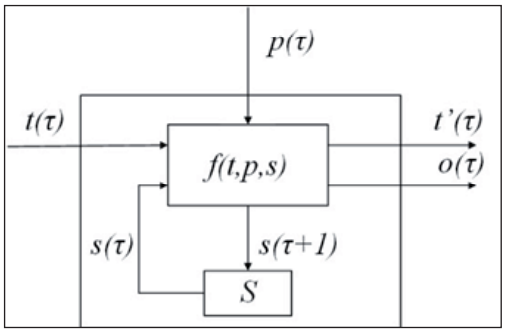

Рис. 2. Автоматная модель ССЗИ

Каждая политика может быть представлена следующей тройкой векторов:

• $\vec{x} = (x_1, x_2, \ldots)$ — входной вектор, описывающий входные параметры ССЗИ. Здесь $x_i \in X_i$, X_i — множество однотипных параметров сетевого трафика (например, IP-адресов, протоколов, номеров портов протоколов транспортного уровня или других характеристик сетевого трафика).

• $\vec{y} = (y_1, y_2, \ldots)$ — выходной вектор, описывающий выходные параметры ССЗИ, то есть сетевой трафик аналогично вектору $\vec{x}$, а также сообщения, генерируемые ССЗИ.

• $\vec{z} = (z_1, z_2, \ldots)$ — вектор состояния, описывающий состояние ССЗИ.

Политика $p \in P$ называется *определенной*, если она описывается тройкой векторов, содержащих определенные в явном виде значения параметров. Политика $p_{gen} \in P_{gen}$ называется *обобщенной*, если параметры соответствующей ей тройки векторов заданы в общем виде без указания конкретных значений (здесь P_{gen} — множество обобщенных политик). Политики множества P являются определенными политиками, так как только определенные политики могут быть реализованы в ССЗИ.

Пусть, например, входной вектор $\vec{x} = (192.168.1.1, TCP, 80)$ принадлежит некоторой определенной политике $p \in P$, тогда входной вектор соответствующей обобщенной политики $p_{gen} \in P_{gen}$ имеет следующий вид: $\overrightarrow{x'} = (Source\ IP, Protocol, Port\ number)$. Каждой обобщенной политике $p_{gen} \in P_{gen}$ соответствует некоторое подмножество $P' \in P$ определенных политик, которые она описывает в общем виде. Пусть P_{sub} — множество подмножеств P, тогда $Def: P_{gen} \to P_{sub}$ является инъективным отображением, ставящим в соответствие каждой обобщенной политике подмножество определенных политик.

Обозначим $X(p)$, $Y(p)$ и $Z(p)$ соответственно входной, выходной и дополнительный векторы обобщенной политики $p \in P_{gen}$. Пусть $\leq$ — отношение частичного порядка на множестве P_{gen} и $p \leq p'$ тогда и только тогда, когда входной, выходной и дополнительный векторы политики p включают параметры соответственно входного, выходного и дополнительного векторов политики p':

$$p \leq p' \leftrightarrow X(p) \subseteq X(p'), Y(p) \subseteq Y(p'), Z(p) \subseteq Z(p').$$

Пусть $\mathcal{F}$ — множество ССЗИ. ССЗИ $F \in \mathcal{F}$ *реализует* политику $p \in P$ (подмно-жество политик $P' \subseteq P$), если p (P') включена в область определения δ и f, обозначение — $f(p)$ ($f(P')$). Каждое ССЗИ $f \in F$ определено на всем множестве T и на некотором подмножестве $P' \subseteq P$, где P' может быть выражено как следующее объединение:

$$P' = Def(p_{gen1}) \cup Def(p_{gen2}) \cup \ldots \cup Def(p_{genk}).$$

ССЗИ $F(Def(p_{gen})) \in \mathcal{F}$ является *простым*, если не существует ССЗИ $F' \in \mathcal{F}$, такого, что его множество значений является подмножеством множества значений F' и вектора p'_{gen} содержат меньше параметров, чем соответствующие вектора p_{gen}:

$$F(D\ e(fp_{gen})) \in \mathcal{F}_s \leftrightarrow \nexists\, F'(D\ e(fp'_{gen})) \in \mathcal{F}\ \forall p'_{gen}: p'_{gen} \le p_{gen}\ \wedge\ p_{gen} \not\geq p'_{gen},$$

где $\mathcal{F}_s$ — множество простых ССЗИ. Другими словами, если исключение из политики любого набора параметров приводит к тому, что политика становится бессмысленной, то такая политика является простой. Пусть множество простых политик $P_s = \{p_{gen} \in P_{gen}: \exists F(Def(p_{gen})) \in \mathcal{F}_s\}$ является множеством таких обобщенных политик, что множество соответствующих им определенных политик может быть реализовано простыми ССЗИ. Если ССЗИ не является простым, то оно является *составным*. Составное ССЗИ может быть построено с использованием простых ССЗИ, которые могут быть соединены последовательно или параллельно. Например, политика МЭ Check Point (рис. 1) и трансляции сетевых адресов (NAT) (рис. 3) представлена как комбинация простых ССЗИ на рис. 4. В силу того, что каждое простое ССЗИ, представляющее функции МЭ, блокирует любой трафик, не соответствующий политике, последнее правило МЭ (рис. 1) не представлено на рис. 4. В отличие от политики МЭ, политика NAT не вносит никаких изменений в трафик, если он не соответствует правилам политики, поэтому на рис. 4 добавлено простое ССЗИ $F_{nat}(default)$, пропускающее трафик без изменений, если он не соответствует двум другим правилам.

NO.	ORIGINAL PACKET			TRANSLATED PACKET			INSTALL ON	COMMENT
	SOURCE	DESTINATION	SERVICE	SOURCE	DESTINATION	SERVICE		
1	Web-server	* Any	* Any	Web-server (Valid Addr	= Original	= Original	Corporate-gw	Automatic rule (see the network object data).
2	* Any	Web-server (Vali	* Any	= Original	Web-server	= Original	Corporate-gw	Automatic rule (see the network object data).

Рис. 3. Пример политики NAT

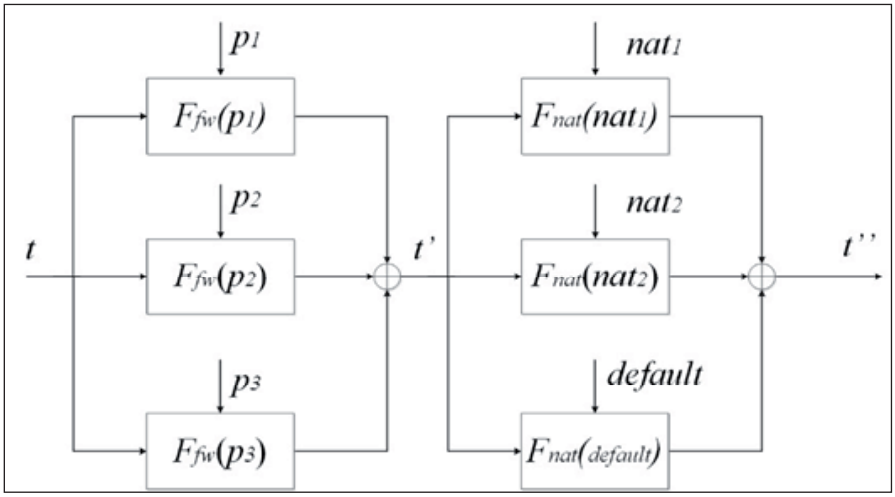

Рис. 4. Пример составного ССЗИ

Эквивалентность ССЗИ

Два простых ССЗИ $F_1(Def(p_{gen1})), F_2(Def(p_{gen2})) \in \mathcal{F}_s$ являются *эквивалентными* тогда и только тогда, когда они генерируют одинаковые выходы при реализации политик:

$$F_1(Def(p_{gen1}) \cong F_2(Def(p_{gen2})) \leftrightarrow \forall p_1 \in Def(p_{gen1})\ \exists p_2 \in Def(p_{gen2}): f_1(t, p_1, s_1^i) = f_2(t, p_2, s_2^j)\ \forall t \in T,$$

где s_1^i и s_2^j — начальные состояния F_1 и F_2 соответственно.

Множество простых ССЗИ F_s разбивается на классы эквивалентности. Все ССЗИ, принадлежащие одному классу эквивалентности, формируют одинаковые выходы при одинаковых входах при реализации соответствующих политик. Однако данные политики могут быть различны с точки зрения синтаксиса.

В качестве демонстрации эквивалентных ССЗИ рассмотрим МЭ Cisco PIX, МЭ Check Point Firewall-1 и систему обнаружения вторжений Snort. Данные ССЗИ обладают различной архитектурой и механизмами реализации политики ИБ. Кроме того, для настройки данных ССЗИ используются различные пользовательские интерфейсы: маршрутизаторы Cisco используют интерфейс командной строки, Check Point Firewall-1 настраивается с помощью графического интерфейса, а Snort не имеет собственного пользовательского интерфейса и настраивается путем изменения конфигурационных файлов. Однако перечисленные ССЗИ, так или иначе, выполняют функцию фильтрации сетевого трафика. Пусть политика ИБ содержит следующее требование:

«Доступ к серверу 10.1.1.10 может осуществляться только из сети 192.168.1.0/24 и только по протоколу http. Все обращения к серверу должны записываться в журнал событий».

Данная политика включает в себя как политику авторизации (доступ к серверу), так и политику на основе обязательств (запись в журнал событий). Политика для МЭ Check Point показана на рис. 5, политика для Cisco представлена в виде команды списка контроля доступа (например, access-list 101):

access-list 101 permit tcp 192.168.1.0 255.255.255.0 host 10.1.1.10 eq 80 log

Политика для Snort имеет следующий вид:

log tcp 192.168.1.0/24 any -> 10.1.1.10 80

Входной вектор обобщенной политики для Check Point, Cisco и Snort имеет вид:

• (Protocol, Source Address, Destination Address, Destination Port);

• (Source Address, Destination Address, Protocol, Destination Port);

• (Protocol, Source Address, Source Port, Destination Address, Destination Port).

Выходной трафик является тем же, что и входной. Это выражено ключевыми словами *accept* и *permit* в случае Check Point и Cisco соответственно. Ключевое слово *log* в обоих случаях обозначает запись в журнал событий. В случае Snort данное ключевое слово также обозначает запись в журнал и то, что выходной трафик совпадает с входным. Выходной вектор для политик: (Accept, Log). Вектор состояний для Check Point и Cisco — (VPN, Time, Comment) и (Access-list Number) соответственно, однако его значения не заданы, что не влияет на политику.

Как видно из приведенных примеров, с точки зрения синтаксиса политики Cisco и Snort различны. Кроме того, в Check Point применяется графический пользовательский интерфейс, что не позволяет сравнить его с Cisco и Snort с точки зрения синтаксиса. Однако семантически эти политики эквивалентны. Очевидно, что ССЗИ Check Point Firewall-1, Cisco PIX и Snort IDS не эквивалентны в целом, но их абстрактные подсистемы (то есть простые ССЗИ, которые реализуют данные политики) являются эквивалентными.

Подход к классификации ССЗИ состоит из следующих шагов:

• Разбиение ССЗИ на простые ССЗИ;

• Определение троек векторов для каждого простого ССЗИ;

• Сравнение векторов с соответствующими векторами существующих классов:

- Если совпадение с существующим классом найдено, то простое ССЗИ принадлежит данному классу;

- Если совпадение не найдено, то простое ССЗИ формирует собственный класс.

Классификация позволяет назначить каждому ССЗИ $F \in \mathcal{F}$ рейтинг R:

$$R_F = \sum_i W_{F_i} I_{F_i}^F,$$

где $W_{F_i} > 0$ — весовой коэффициент простого ССЗИ F_i, который может быть вычислен методом экспертных оценок, и $I_{F_i}^F \in \{0,1\}$ — индикатор, который показывает, включено ли ССЗИ F в класс эквивалентности F_i.

NO.	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
1	Network-192.168.1.0_24	Server-10.1.1.10	Any Traffic	http	accept	Log	Corporate-gw	Any	

Рис. 5. Правило политики для МЭ Check Point

Классификация ССЗИ позволяет выбрать ССЗИ с требуемой функциональностью. Например, если необходимо ССЗИ с функциями F_a, F_b и F_c в некотором узле сети для реализации некоторой политики, то для этого ССЗИ должно быть одновременно включено в классы F_a, F_b и F_c. Если таких ССЗИ несколько, то наиболее эффективным является то, которое имеет максимальное значение отношения рейтинга к цене:

$$\frac{R_F}{C_F} \to max,$$

где C_F — стоимость ССЗИ F. Если не существует ССЗИ, входящего в требуемые классы, то могут быть выбраны комбинации ССЗИ, включающие требуемые функции, и вычислены суммарные рейтинги для каждой комбинации:

$$\sum_j \frac{R_{F^j}}{C_{F^j}} \to max,$$

где $j \in \{j_1, j_2, \dots, j_k\}$ и $\{F^{j_1}, F^{j_2}, \dots, F^{j_k}\}$ — множество ССЗИ, входящих в комбина-цию.

Простые ССЗИ и их комбинации могут быть представлены посредством многосортной алгебры. Алгебра может быть преобразована в формальный язык, заданный порождающей грамматикой. Представление политик как формального языка позволяет применять к ним алгоритмы синтаксического анализа, а также осуществить трансляцию политик в конфигурации ССЗИ. Описание алгебры и формального языка для представления политик выходит за рамки данной статьи.

Заключение

Основными операциями в рамках процесса управления политиками для ССЗИ являются пересмотр существующих политик и выявление конфликтов в них, разработка новых политик и требований к ССЗИ, выбор ССЗИ и их внедрение в существующую СОИБ, а также реализация политик. С ростом числа ССЗИ и увеличением их сложности процесс становится трудоемким и подверженным ошибкам.

Применение формальных подходов к моделированию политик способствует автоматизации данных операций процесса. Существующие на сегодняшний день подходы в большинстве случаев позволяют формализовать политики и транслировать их в конфигурацию ССЗИ, а также выявлять конфликты в политиках. Однако эти подходы либо не позволяют задавать весь спектр политик для ССЗИ, либо не поддерживают возможность нахождения ССЗИ, соответствующих определенным требованиям и способных реализовать разработанные политики.

Представленная в статье модель основана на классификации ССЗИ по их функциональным возможностям и позволяет формализовать ССЗИ и реализуемые ими политики (как политики авторизации, так и политики на основе обязательств). Политики в рамках модели задаются независимо от платформы ССЗИ, в которой они будут реализованы. Кроме того, модель позволяет

осуществить выбор наиболее эффективного ССЗИ для реализации политик. Дальнейшим развитием модели является разработка методов выявления и разрешения конфликтов в политиках.

СПИСОК ЛИТЕРАТУРЫ:

1. 2014 Cyberthreat Defense Report North America & Europe. CyberEdge Group. 2014.
2. SANS Survey on Network Security Results 2013. SANS 2013.
3. The State of Network Security 2013: Attitudes and Opinions. AlgoSec 2013.
4. *Chapple M. J., D'Arcy J., Striegel A.* An Analysis of Firewall Rulebase (Mis)Management Practices // ISSA Journal. February 2009. P. 12–18.
5. *Wool A.* Trends in Firewall Configuration Errors: Measuring the Holes in Swiss Cheese // IEEE Internet Computing. 2010. Vol. 14. Issue 4. P. 58–65.
6. *Rees J., Bandyopadhyay S., Spafford E. H.* PFIRES: a policy framework for information security // Communications of the ACM. 2003. Vol. 46. Issue 7. P. 101–106.
7. *Wahsheh L. A., Alves-Foss J.* Security Policy Development: Towards a Life-Cycle and Logic-Based Verification Model // American Journal of Applied Sciences. 2008. 5 (9). P. 1117–1126.
8. *Knapp K. J., Morris R. F. Jr., Marshall T. E., Byrd T. A.* Information security policy: An organizational-level process model // Computers & Security. 2009. 28. P. 493–508.
9. *Tuyikeze T., Pottas D.* An Information Security Policy Development Life Cycle // Proceedings of the South African Information Security Multi-Conference (SAISMC). 2010. P. 241–266.
10. *Laborde R., Kamel M., Barrure F., Benzekri A.* Implementation of a formal security policy refinement process in WBEM architecture // Proceeding of 4th Latin American Network Operations and Management Symposium. 2005. P. 241–266.
11. *Cuppens F., Cuppens-Boulahia N., Sans T., Miuge A.* A Formal Approach to Specify and Deploy a Network Security Policy // Proceeding of Second Workshop on Formal Aspects in Security and Trust. 2004. P. 203–218.
12. *Preda S., Cuppens-Boulahia N., Cuppens N., Garcia-Alfaro J., Toutain L.* Model-Driven Security Policy Deployment: Property Oriented Approach // Proceedings of the Second International Symposium on Engineering Secure Software and Systems. 2010. P. 123–139.
13. *Preda S., Cuppens-Boulahia N., Cuppens F., Toutain L.* Architecture-Aware Adaptive Deployment of Contextual Security Policies // Proceedings of International Conference on Availability, Reliability, and Security. 2010. P. 87–95.
14. *Garcia-Alfaro J., Cuppens F., Cuppens-Boulahia N., Preda S.* MIRAGE: A Management Tool for the Analysis and Deployment of Network Security Policies // Proceedings of the 5th International Workshop on Data Privacy Management and 3rd International Conference on Autonomous Spontaneous Security. 2010. P. 203–215.
15. *Craven R., Lobo J., Lupu E., Russo A., Sloman M.* Security Policy Refinement Using Data Integration: a position paper // Proceedings of the 2nd ACM workshop on Assurable and Usable Security Configuration. 2009. P. 25–28.
16. *Craven R., Lobo J., Lupu E., Russo A., Sloman M.* Decomposition Techniques for Policy Refinement // Proceedings of the 6th International Conference on Network and Service Management. 2010. P. 72–79.
17. *Zhao H., Lobo J., Roy A., Bellovin S. M.* Policy Refinement of Network Services for MANETs // Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management. 2011. P. 113–120.
18. *Sonnenbichler A., Geyer-Schulz A.* ADQL: A Flexible Access Definition and Query Language to Define Access Control Models // Proceedings of the 9th International Conference on Security and Cryptography. 2012. P. 379–386.
19. *Crampton J., Morisset C.* Towards a generic formal framework for access control systems // The Computing Research Repository. Vol. abs/1204.2342. 2012.
20. *Casalino M. M., Thion R.* Refactoring Multi-layered Access Control Policies Through (De)composition // Proceedings of the 9th International Conference on Network and Service Management. 2013. P. 243–250.

REFERENCES:

1. 2014 Cyberthreat Defense Report North America & Europe. CyberEdge Group. 2014.
2. SANS Survey on Network Security Results 2013. SANS 2013.
3. The State of Network Security 2013: Attitudes and Opinions. AlgoSec 2013.
4. *Chapple M. J., D'Arcy J., Striegel A.* An Analysis of Firewall Rulebase (Mis)Management Practices // ISSA Journal. February 2009. P. 12–18.
5. *Wool A.* Trends in Firewall Configuration Errors: Measuring the Holes in Swiss Cheese // IEEE Internet Computing. 2010. Vol. 14. Issue 4. P. 58–65.
6. *Rees J., Bandyopadhyay S., Spafford E. H.* PFIRES: a policy framework for information security // Communications of the ACM. 2003. Vol. 46. Issue 7. P. 101–106.
7. *Wahsheh L. A., Alves-Foss J.* Security Policy Development: Towards a Life-Cycle and Logic-Based Verification Model // American Journal of Applied Sciences. 2008. 5 (9). P. 1117–1126.

8. *Knapp K. J., Morris R. F. Jr., Marshall T. E., Byrd T. A.* Information security policy: An organizational-level process model // Computers & Security. 2009. 28. P. 493–508.
9. *Tuyikeze T., Pottas D.* An Information Security Policy Development Life Cycle // Proceedings of the South African Information Security Multi-Conference (SAISMC). 2010. P. 165–176.
10. *Laborde R., Kamel M., Barrure F., Benzekri A.* Implementation of a formal security policy refinement process in WBEM architecture // Proceeding of 4th Latin American Network Operations and Management Symposium. 2005. P. 241–266.
11. *Cuppens F., Cuppens-Boulahia N., Sans T., Miuge A.* A Formal Approach to Specify and Deploy a Network Security Policy // Proceeding of Second Workshop on Formal Aspects in Security and Trust. 2004. P. 203–218.
12. *Preda S., Cuppens-Boulahia N., Cuppens N., Garcia-Alfaro J., Toutain L.* Model-Driven Security Policy Deployment: Property Oriented Approach // Proceedings of the Second International Symposium on Engineering Secure Software and Systems. 2010. P. 123–139.
13. *Preda S., Cuppens-Boulahia N., Cuppens F., Toutain L.* Architecture-Aware Adaptive Deployment of Contextual Security Policies // Proceedings of International Conference on Availability, Reliability, and Security. 2010. P. 87–95.
14. *Garcia-Alfaro J., Cuppens F., Cuppens-Boulahia N., Preda S.* MIRAGE: A Management Tool for the Analysis and Deployment of Network Security Policies // Proceedings of the 5th International Workshop on Data Privacy Management and 3rd International Conference on Autonomous Spontaneous Security. 2010. P. 203–215.
15. *Craven R., Lobo J., Lupu E., Russo A., Sloman M.* Security Policy Refinement Using Data Integration: a position paper // Proceedings of the 2nd ACM workshop on Assurable and Usable Security Configuration. 2009. P. 25–28.
16. *Craven R., Lobo J., Lupu E., Russo A., Sloman M.* Decomposition Techniques for Policy Refinement // Proceedings of the 6th International Conference on Network and Service Management. 2010. P. 72–79.
17. *Zhao H., Lobo J., Roy A., Bellovin S. M.* Policy Refinement of Network Services for MANETs // Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management. 2011. P. 113–120.
18. *Sonnenbichler A., Geyer-Schulz A.* ADQL: A Flexible Access Definition and Query Language to Define Access Control Models // Proceedings of the 9th International Conference on Security and Cryptography. 2012. P. 379–386.
19. *Crampton J., Morisset C.* Towards a generic formal framework for access control systems // The Computing Research Repository. Vol. abs/1204.2342. 2012.
20. *Casalino M. M., Thion R.* Refactoring Multi-layered Access Control Policies Through (De)composition // Proceedings of the 9th International Conference on Network and Service Management. 2013. P. 243–250.