В. С. Горбатов, А. Ф. Лобанов

ОБЕСПЕЧЕНИЕ ЗАЩИТЫ ОТКРЫТЫХ СРЕД НА ПРИМЕРЕ АРХИТЕКТУРЫ SNA

Введение

Развитие рынка ИТ-технологий идет по двум диалектически противоположным направлениям. Первое — очевидное и традиционное стремление к использованию достижений научно-технического прогресса (изобретений, ноу-хау, инноваций) с целью увеличения доходов за счет роста нормы прибыли путем удовлетворения новых уникальных потребностей пользователей. В противовес этому фактору бурное распространение компьютерных систем во все сферы производственной и общественной деятельности привело в настоящее время к преобладанию второго направления, связанного с решением вопросов совместимости различных технологий и систем, их унификации, стандартизации и т. д. В этом случае увеличение доходов обеспечивается за счет увеличения серийности поставок техники и услуг, что ведет к соответствующему росту прибыли.

Примером влияния рассмотренных факторов является использование широко известных сетевых стратегий по практической реализации ИТ-технологий конца 80-х годов прошлого века, на основе которых строились вычислительные среды: SNA (System Network Architecture) фирмы IBM и DNA (Digital Network Architecture) фирмы DEC. Возрастание числа вычислительных сред различной функциональной направленности, а также количества объединяемых с помощью этих сетевых архитектур вычислительных комплексов выявило основной технологический недостаток разработки систем на основе исключительно первого направления. Для обеспечения конкурентоспособности создание вычислительных сред на начальных этапах проводилось исходя из принципа сугубой «закрытости» разработок. И, как следствие, появилась явная несовместимость разработанных сред между собой. Это дало толчок к появлению концепции разработки так называемых открытых систем (вычислительных сред), одним из основных принципов которой стало объединение усилий различных групп разработчиков под эгидой негосударственных международных организаций. Наиболее известными из них являются X/Open; OSF (Open Software Foundation); IEEE (Institute of Electrical and Electronics Engineers).

Каждая из этих организаций имеет свое понимание понятия «открытые системы»:

X/Open — это приверженность стандартам при соблюдении независимости производителей систем, устройств и средств ИТ-технологий;

OSF под открытой системой понимает комплексы, которые имеют стандартный набор интерфейсов для организации взаимной работы, переносимость программных продуктов и защиту инвестиций по этим работам;

IEEE определяла открытость как соответствие международным стандартам с выполнением взаимной работы, переносимость прикладных программ, данных и пользователей.

Тем не менее по совместной договоренности в качестве основных характеристик в стандартах открытости были приняты возможность переносимости и взаимной работы (интероперабельность).

При введении такого понятия открытости систем и сред неизмеримо выросла роль стандартов, определяющих набор правил, применяемых к объектам. Набор стандартов представляет собой определенную спецификацию моделей (например, модель OSI — Open Standard Interconnections). Принято выделять следующие основные типы стандартов: стандарты de jure и de facto. На высшем уровне стандарты de jure разделяются на стандарты международные и национальные.

Ведущая организация в области разработки открытых систем — X/Open — была создана в 1984 г. С этого момента начинается общепризнанное определение базовых элементов для построения прикладных программ, совместимых на уровне исходных кодов. Эти прикладные программы имеют свойства переносимости, коммуникабельности, а также имеют интерфейс с системой.

Ключевая роль OSF на этом этапе состояла в том, что в рамках концепции открытых систем были разработаны базовые стандарты операционных систем, графический пользовательский интерфейс, стандарты распределенной обработки DCE (Distribution Computing Environment) и DME (управление распределенной средой), спецификации среды прикладных программ (AES).

Опираясь на эти и другие рекомендации, к середине 90-х годов XX в. ведущие фирмы — разработчики ИТ-технологий представили свои программы создания аппаратно-программных средств на основе стандартов открытых систем.

Очевидно, что подобный подход может быть применим и для исследуемой нами области информационной безопасности. Ярким примером этому являются разработки фирмы IBM как наиболее представительного игрока на рынке ИТ-технологий.

1. Сетевая архитектура SNA (основные сведения)

1.1. Начальный этап разработки

В 80-х годах прошлого века основу вычислительных сред составляли большие компьютеры (mainframes). Они имели значительное число терминалов пользователей, построенных на базовой структуре 3270. В состав операционных систем этих ЭВМ входил, на правах отдельной ОС, программный интерфейс VTAM (virtual telecommunication acts method), который поддерживал связи с удаленными терминалами и другими системами. В таком «классическом» виде архитектура SNA становится стандартом de facto для информационной индустрии, причем ее можно рассматривать для конечных точек (терминалов) как открытую архитектуру, позволяющую относительно легко вводить дополнительные функции в систему.

В конце 1980-х годов в архитектуру SNA были введены в качестве транспортных средств пакетные сети X.25, которые использовались для организации связи с терминалами, а также персональные компьютеры (PC), исполняющие роль конечных терминалов. Все PC имели свои операционные системы PC-DOS и MS-DOS, разработанные на базе ОС CMS, которая, в свою очередь, была разработана в рамках ОС VM и предназначалась для работы с терминалами пользователей. В такие новые терминалы на базе PC сразу же закладывались определенные самостоятельные вычислительные возможности. Эти PC имели три режима работы:

- режим стандартного терминала большой ЭВМ (под управлением ОС CMS);
- режим эмуляции терминала (под управлением эмулятора, построенного на базе ОС PC-DOS);
- автономный режим работы под управлением ОС PC-DOS или MS-DOS.

При появлении локальных сетей (например, Token-Ring, Ethernet) они были использованы для расширения коммуникационных средств связи персональных компьютеров с большими

машинами. К слову сказать, эти сети и разрабатывались с учетом такого опыта использования терминалов. Например, в стратегии DNA фирмы DEC сети рассматривались в основном именно в режиме подключения терминалов и других устройств к общей системе.

Архитектура SNA организует работу с помощью специальных форматов и протоколов. Все устройства в этой архитектуре рассматриваются как логические (LU) и физические (PU) устройства. Существует семь различных типов логических устройств. Организация работы связана с типом устройства и типом потока данных в сеансе. Типы потоков в архитектуре соотносятся с типом терминала, которых в «классической» архитектуре было два — 3270 и 5250. Различают в архитектуре два типа сеансов — служебные и информационные (LU — LU). Работа строится на базе организации сеансов LU — LU.

Управление в сетевой архитектуре строится с помощью следующих продуктов:

-VTAM — телекоммуникационный метод доступа;

- NCP — программа управления сетью;

- подсистемы прикладных программ;

- программы мониторинга (Netview), которые отвечают за выявление ошибок в системе и администрирование.

1.2. Развитие сетевой архитектуры SNA

1.2.1. Организация режима «точка — точка»

Форматы и протоколы, разработанные в архитектуре SNA, получили свое развитие в протоколах APPC (Advanced Program-to-Program Communication), которые развивают организацию работы сеансов LU — LU на базе нового логического устройства LU 6.2. Основная задача этих протоколов APPC — организация связи типа «точка — точка» между периферийными устройствами.

Главная концепция APPC — организация связи между двумя программами в сеансе в диалоговом режиме. Обмен данных производится по сети. В этом интерфейсе APPC вводятся параллельные сеансы, которые используют следующие коммуникационные средства:

- увеличение полосы пропускания между двумя локальными устройствами;

- баланс загрузки линий связи (физических каналов);

- возможности управления разными сеансами.

Логическое устройство LU 6.2, введенное в архитектуру, поддерживает работу новых типов терминалов, построенных на PC и локальных сетях (физические устройства PU 2.1), и имеет следующие базовые режимы работы:

- передача данных;

- вызов удаленных физических устройств;

- управление потоком;

- работа с ошибками;

- синхронизация процессов.

Логическое устройство LU 6.2, кроме того, позволяет выполнять следующие операции размещения (mapped):

- совместимость управляющих заголовков;

- расположение данных;

- управление потоками данных;

- обнаружение ошибок.

Таким образом, переход к новым типам терминалов (локальные сети — PU 2.1) привел к неизбежному развитию архитектуры SNA в части перехода к сетевой организации по признаку работы равных (peer). Такая организация работы была основана на ключевых компонентах: APPC,

последующих продуктах типа APPN (Advanced Peer-to-Peer Networking Architecture) и сетевых программных интерфейсов CPI-C. На базе развития APPC – LU 6.2 и PU 2.1 была создана основа для перехода на протоколы TCP/IP.

1.2.2. Создание многопротокольной сети

Протокол APPN привел к организации архитектуры APPI (Advanced Peer-to-Peer Internetworking), которая позволила организовать передачу данных по графику протокола IP. Протокол APPN и архитектура APPI позволили создать архитектуру сети с сетевым сервисом, где стали возможны такие режимы работы, как динамическая реконфигурация сети и динамическое расположение пользователей.

В новой сетевой архитектуре возможно введение узлов, которые выполняют следующие функции:

- организация серверов, работающих с директориальным сервисом;
- организация серверов с сеансовой поддержкой;
- организация сервисов управления сетью;
- введение клиентов в конечных узлах.

В результате разработки и введения в архитектуру SNA перечисленных архитектур и протоколов была создана возможность организации многопротокольных сетей на базе протокола IP. Многопротокольная организация работы сетевой архитектуры привела к появлению и использованию следующих устройств:

- многопротокольные рутеры (router), предназначенные для выделения и организации маршрутов большинства действующих протоколов – TCP/IP, DECnet, XNS, IPX, SNA, NETBIOS;
- сетевые процессоры и мосты (bridges), ориентированные на скорость 2048 Мбит/с, которая позволила работать с модемами по интерфейсу V.35 (на скоростях до 64 Кб/с) и интерфейсу RS – 422 (на скоростях до 1,536 Мб/с).

Была усовершенствована также работа функционального управления архитектуры SNA путем введения следующих средств и сервисов:

- программные менеджеры для управления логическими и физическими узлами в части инициирования и управления сеансами работы;
- управление функциями сервисов конечных пользователей и сервисов сетевых сеансов;
- функциональное управление данными.

2. Основные подходы по организации защиты в SNA

2.1. Уровневая модель

Архитектура SNA была построена по уровневому принципу, причем уровни SNA во многом совпадают с функциональными назначениями уровней открытой семиуровневой модели ISO. Однако есть и существенные расхождения в функциональных назначениях уровней этих моделей, обусловленные спецификой архитектуры и направленности развития режимов работы. Например, второй уровень SNA был вначале ориентирован на протокол последовательного интерфейса SDLC. В открытой модели ISO этот уровень «Data link» ориентирован на поддержку типов различных сетей, сетевых протоколов. Ниже приведена общая схема защиты информации в модели SNA.

Сервисы защиты	Уровни модели SNA
1. Целостность информации:	
- передача наборов информации (frames)	1, 2
- ошибки маршрутизации	3
- ошибки в потоке информации	5

2. Конфиденциальность:	
- шифрование данных	4
- проверка контрольных сумм	4
3. Идентификация сообщений:	
- о последовательности сообщений	3
4. Идентификация и аутентификация	6
5. Доступ к сервису	6
6. Идентификация конфигурации	6
7. Проверка пути	На разных уровнях
8. Проверка авторизации	,,
9. Проверка имени класса сервиса	,,

Уровень протокола SDLC (в открытой модели ISO соответствует 1-му и 2-му уровням) предназначен для формирования передаваемого сообщения (в терминах SNA эти сообщения называются Frames) и передачи его в конечную точку (терминал). Это сообщение содержит следующие заголовки полей информации:

- поле адреса;
- поле управляющей информации (три типа формата, которые предназначены для идентификации сообщения-запроса, для указания отсутствия информации в сообщении, идентификатор разных типов команд);
- поле информации;
- поле для контрольной суммы (CRC — Cyclic redundancy check) размером в 16 бит.

Проверка CRC производится на обнаружение ошибки. Эта проверка производится по всем битам передаваемой посылки, включая поле адреса, поле управляющей информации и информационное поле. При обнаружении ошибки вторичная станция передает сообщения в первичное (исходное) устройство. Первичное устройство заканчивает передачу и выдает сообщение о получении готовности. Вторичная станция устанавливает готовность для повторного приема посылки.

Ниже показано соответствие уровней модели SNA и модели OSI.

Модель SNA		OSI
6-й уровень	Функции управления	уровень представления
5-й уровень	Управление потоком данных	уровень сеансный
4-й уровень	Управление передачами	уровень транспортный
3-й уровень	Управление путем	уровень сетевой
2-й уровень	Управление установлением связи	уровень управления связи
1-й уровень	Физический уровень	уровень физический

Был принят следующий порядок подготовки пакетов сообщений в модели SNA:

- на уровне 5 и 6 подготавливается информация о запросе и ответе (R/R unit);
- на транспортном уровне формируется базовая информационная единица (BIU) с заголовком RH;

- на уровне управления путем формируется передаваемая единица BTU (добавляется к базовой информационной единице заголовок передачи);
- на уровне установления связи к пакету данных добавляется заголовок протокола SDLC.

2.2. Совместимость разных ОС в единой сетевой среде

Все операционные системы фирмы IBM имеют средства защиты, причем для всех классов машин защита построена на принципе разграничения доступа к ресурсам отдельного пользователя. Для различных терминалов системы существовали права на получение системных ресурсов, которые предоставлялись администратором системы. Доступ в систему производился по персональному идентификатору и проверке на индивидуальный пароль. Доступ пользователей к системным ресурсам жестко ограничивался. Операционная система VM, например, была спроектирована таким образом, что конкретный пользователь имел свою виртуальную машину с выделенными ему ресурсами, далее которых он не имел доступа, кроме общих ресурсов, например к внешней памяти, к принтеру и т. д. Вся система жестко контролировалась администратором, и постоянно проводился мониторинг работы пользователей и ресурсов системы. Постоянно (по выбранному графику) проводилось дампирование (снятие резервных копий) системных и пользовательских файлов.

Развитие технологий и модернизация модели SNA приводят к необходимости обеспечения совместной работы разных систем сначала в рамках модели SNA. Это потребовало и работ по модернизации средств защиты. В частности, было разработано средство RACF (Resource Access Control Facility), которое работало на всех ОС фирмы. Основными функциями RACF были введение поддержки идентификации пользователей, проверки и авторизации доступа к данным. Основной функцией RACF является введение:
- базовых свойств защиты;
- пользовательского интерфейса;
- общей процедуры доступа к ресурсам;
- средств защиты отдельных PC и локальных сетей;
- администратора защиты с интерфейсом администратора и набором команд для работы с защитой.

Базовые средства защиты включали в свой состав следующие функции:
- дискретное управление доступом (DAC), которое проводилось через профили дискретного подхода к ресурсам собственника;
- мандатное управление доступом (MAC), основанное на комбинации уровней защиты и введении категорий пользователей в части прав. При этом ресурсам и пользователям присваивались метки защищенности.

Пользователь имел доступ к интерфейсу в соответствии паролем. При использовании этого интерфейса применялась криптографическая защита по алгоритму, разработанному на соответствие рекомендаций стандарта DES.

Процедура доступа к ресурсам производилась через операционные системы фирмы: MVS, CICS, TSO и с помощью интерфейса виртуальных машин. Запрос передавался через средство RACF. Авторизация доступа проводилась на четырех уровнях защиты. Реализация старта этой процедуры начиналась с таблицы глобального доступа.

Отдельные PC и локальные сети, входящие в состав отдельной системы, включаются в конфигурацию средства RACF в том случае, если они поддерживают логическое устройство LU 6.2 и средство APPC/MVS при серверной поддержке.

Средство RACF управляет набором профилей. Это позволяет определять системных пользователей и их доступ к ресурсам. Пользователи рассматриваются в следующем порядке:

- профили пользователей, куда включаются базовые сегменты (имя пользователя, уникальный идентификатор, пароль, идентификатор пользователя и другие атрибуты); другие сегменты, связанные со средствами TSO, DFP, CICS, языки программирования и процедурные языки;
- профили групп пользователей, в которые могут входить собственные ресурсы, пользователи, другие группы и подгруппы;
- профили основных ресурсов, которые содержат список доступа, список групп, опции аудита, список пользователей, уровень защиты и категории для ответа; RACF имеет 48 классов ресурсов, в которые входят и накопители на дисках, лентах, системы транзакций и IMS;
- собственники, или оригинально идентифицированные пользователи в системе, которые имеют свои профили со своим доступом и т. д.

Средство RACF поддерживается следующими коммуникационными возможностями:
- телекоммуникационной системы доступа (VTAM);
- архитектуры SNA;
- коммуникационных средств LU 6.2;
- коммуникационных средств APPC/MVS (на базе ОС MVS).

RACF имеет следующие расширения:
- открытая редакция ОС MVS, которая разработана с учетом требований стандартов POSIX, авторизации пользовательских прикладных программ и защитысредствами программного управления Netview;
- поддержка APPC/MVS, которая расширяет число основных классов ресурсов, автоматизирует и расширяет консольные средства для Netview;
- различные версии RACF.

Заключение

Из приведенного достаточно краткого обзора видно, как фирма IBM постепенно двигается в сторону развития своей сетевой стратегии SNA, стремясь организовать работу этой сетевой архитектуры на соответствие рекомендациям и требованиям открытых стандартов, одновременно сохраняя функциональные возможности «первоначальной» сетевой архитектуры. Эти тенденции сохраняются и при разработке средств защиты. Наиболее важные шаги по реализации такой стратегии:
- организация режима работы «равный с равным», который предоставляет большую гибкость в рамках большой сетевой системы;
- создание многопротокольной сети, что позволяет использовать в работе сетевой архитектуры кроме оригинальных протоколов сетевого и транспортного уровней работу протоколов TCP/IP. Этот шаг открывает большие возможности сетевой архитектуры SNA для совместной работы, например, с глобальной сетью Интернет;
- развитие уровневой организации SNA до рекомендаций и требований семиуровневой модели организации ISO позволяет обеспечить совместимость систем различных производителей.

Такой практический опыт может быть полезен отечественным разработчикам ИТ-технологий, а также студентам и слушателям, которые специализируются в области обеспечения безопасности информации в ключевых системах информационной инфраструктуры.