K.A. Butsik, V.V. Dereviashko

The Using of «Time Intervals» for Provision a Secure Network Boot of Thin-Client System

Key words: thin-client, terminal operating system, trusted (secure) network boot, time intervals, stages of boot.

In this article reveals the notion of «time slots» (T) that have been introduced in description of the model and concept of trusted network booting for a thin-client system [1]. The article deals with stages of loading the terminal operating system into workstation memory, starting of workstation switch-on. «Time slot» defines a time (sec.) of execution at each stage in cases of normal operation and insider activity. The article presents methods for the detection and neutralization of such activity by monitoring of values of «time slots».

К.А. Буцик, В.В. Деревяшко

ИСПОЛЬЗОВАНИЕ «ВРЕМЕННЫХ ИНТЕРВАЛОВ» В СИСТЕМЕ ДОВЕРЕННОЙ СЕТЕВОЙ ЗАГРУЗКИ ТОНКОГО КЛИЕНТА

В ходе разработки концепции и модели доверенной сетевой загрузки тонкого клиента, представленной в [1], было введено понятие «временных интервалов» (Т). Метод обеспечения доверенной сетевой загрузки в свою очередь предполагал контроль представленных «временных интервалов» за счет сравнения их нормированных значений со значениями, рассчитанными по факту получения сетевыми устройствами ключевых запросов в автоматическом режиме. Ограничения модели и концепции доверенной сетевой загрузки тонкого клиента от воздействий внутреннего нарушителя сводились к реализации сетевой загрузки терминальной операционной системы (ТОС) с использованием технологии PXE. Рассмотрим этапы работы рабочей станции (РСТТК) на базе типовой ПЭВМ [2] в рамках означенных ограничений, позволяющих выделить необходимые «временные интервалы» и ключевые запросы, подвергаемые контролю:

1) включение питания рабочей станции. Самотестирование напряжений и тока на материнской плате, подача питания и сброс регистров процессора;

2) обращение процессора к ПЗУ BIOS с целью получения POST-программы. Исполнение POST-программы (основные функции):

- копирование BIOS в оперативную память;
- настройка видеоадаптера с выводом сообщений POST;
- инициализация портов, дисководов и устройств;
- инициализация других устройств, содержащих свой BIOS;

3) передача микрокода PXE в оперативную память и процессор. Инициализация сетевой карты и настройка поддержки стека протоколов TCP/IP;

4) получение IP-адреса и базовых параметров сети посредством отправки broadcast-запроса в доступное сетевое пространство (поиск DHCP-сервера) ;

5) отправка broadcast-запроса с целью получения IP-адреса PXE-сервера (TFTP-сервера) ;

6) установка соединения с PXE-сервером и загрузка PXE-файла в оперативную память рабочей станции;

7) управление передается программе, распакованной из PXE-файла. Программа производит подключение к серверу сетевой загрузки (ССЗ) с целью получения файл-образа терминальной операционной системы;

8) для каждой автоматизированной системы на базе технологии «тонкий клиент» (АСТТК) процесс загрузки, распаковки, обработки и исполнения ТОС является уникальным в зависимости от технологической реализации. Окончание этапа – запуск ТОС в памяти рабочей станции;

9) подключение ТОС к терминальному серверу – начало терминальной сессии. РСТТК и сервер терминалов переходят в режим ожидания действий пользователя.

Каждый из представленных этапов характеризуется некоторым временем исполнения – представим его параметром t с порядковым номером от 1 до 9. Необходимо учесть, что при новом включении РСТТК текущее значение t не всегда совпадает со значением, зафиксированным при прошлом включении. Основной причиной этого являются непрогнозируемые флюктуации компонент рабочей станции, серверов и каналов связи [2]. То есть значение «временного интервала» должно включать в себя отклонение $t \in [0; \infty)$, сек – задержку времени исполнения одного или нескольких этапов, попадающих в рассматриваемый «временной интервал». При этом $t = 0$ в случае отсутствия задержки времени, и $t = \infty$ в случае аварийного останова работы РСТТК.

Рассмотрим АСТТК, в которой надежность работы компонент в составе рабочих станций, транспортной сети и серверов не вызывает сомнений. Тогда параметр t не может быть равен ∞, т.е. отсутствует принципиальная возможность аварийного останова (сбоя) какого-либо из элементов системы. В свою очередь допустимо пренебречь задержками времени исполнения ($t = 0$) на этапах 1 и 3, поскольку отклонение времени исполнения для них крайне мало [3] при условии корректной работы компонент РСТТК (блока питания, материнской платы, сетевой карты и т.д.).

Разделим этапы работы РСТТК в соответствии с порядком отправки запросов к элементам АСТТК:

– запросы к DHCP, TFTP и PXE серверам;

– запрос к ССЗ на загрузку ТОС;

– запрос к серверу терминалов на создание терминальной сессии.

Каждый из представленных запросов возможно контролировать доверенными методами на стороне коммутационного оборудования и (или) серверного сегмента. Следовательно, с позиции времени исполнения этапов работы РСТТК допустимо рассматривать следующие «временные интервалы»:

1) отрезок времени с момента включения рабочей станции до отправки первого broadcast-запроса к DHCP (TFTP) серверу. В этот «временной интервал» попадают этапы 1 – 4;

2) отрезок времени с момента получения IP-адреса до отправки запроса на загрузку PXE-файла. В этот «временной интервал» попадают этапы 5 и 6;

3) отрезок времени с момента начала загрузки ТОС в память РСТТК до создания первой терминальной сессии с сервером терминалов (этапы 7 – 9).

Выразим результирующие значения T для каждого ключевого «временного интервала» с использованием описанных выше параметров t и t:

$$T_1 = t_1 + t_2 + \quad t_2 + t_3 + t_4 + \quad t_4 \text{ , сек;} \tag{1}$$

$$T_2 = t_5 + \quad t_5 + t_6 + \quad t_6 \text{ , сек;} \tag{2}$$

$$T_3 = t_7 + \quad t_7 + t_8 + \quad t_8 + t_9 + \quad t_9 \text{ , сек.} \tag{3}$$

Успешность проведения атаки внутренним нарушителем однозначно определяется его возможностью выйти за рамки установленного технологического процесса обработки информации [4]. Ниже приведены сценарии атак, в которых контроль «временных интервалов» устраняет указанную возможность нарушителя. Под контролем понимается ожидание исполнения штатных этапов и сравнение их параметров с нормированными значениями.

1. Попытки нарушителя провести атаку в отсутствие эмуляции штатной загрузки ТОС автоматически исключают один или несколько штатных этапов работы: запросы к DHCP, TFTP или PXE-серверу, получение файл-образа ТОС от ССЗ и т.п. [3]. С позиции контроля «временных интервалов», $T_1, T_2 \to \infty$ и, следовательно, $T_1 + T_2 > T_{1\text{норм}} + T_{2\text{норм}}$.

2. Первый вариант эмуляции штатной загрузки – использование нарушителем съемного носителя для загрузки рабочей станции под управлением нештатной ТОС. Начало атаки требует настройки параметров загрузочной области в ПО BIOS (UEFI) и совпадает с началом 3-го этапа штатной загрузки рабочей станции. При равных прочих отклонение t_3 неизбежно превысит нормированное значение, и, следовательно, $T_1 > T_{1\text{норм}}$. Заранее предположим, что настройка ПО BIOS не требует от нарушителя затрат времени – к примеру, в случае использования не декларированных возможностей конкретного ПО BIOS [5]. Тогда решающим фактором становится время инициализации съемного носителя и загрузки нештатной ОС до момента генерации broadcast-запросов ее средствами включительно. Однако поскольку промежуток времени t_3 (сек) крайне мал, то $t_3 + \ t_3 > t_{3\text{норм}}$, тогда $T_1 > T_{1\text{норм}}$.

3. Второй вариант эмуляции штатной загрузки – использование нарушителем нештатного сервера сетевой загрузки, заранее внедренного в сеть АСТТК. Начало атаки совпадает с началом 4-го этапа штатной загрузки, соответственно $t_4 + \ t_4 = t_{4\text{норм}} + + \ t_{4\text{норм}}$, поскольку broadcast-запрос будет передан к обоим серверам сетевой загрузки (штатному и нештатному). Следовательно, $T_1 = T_{1\text{норм}}$ и, аналогично, $t_5 + \ t_5 = t_{5\text{норм}} + + \ t_{5\text{норм}}$. Однако на этапе 6 нарушителю необходимо вначале получить образ нештатной ОС и затем эмулировать загрузку штатной ТОС в память рабочей станции [3]. Что означает $t_6 + \ t_6 > t_{6\text{норм}} + \ t_{6\text{норм}}$и, следовательно, $T_2 > T_{2\text{норм}}$.

Таким образом, системе защиты достаточно признать представленные выше результаты событием НСД, заблокировать любую сетевую активность указанной рабочей станции и выслать уведомление администратору безопасности информации.

Следует отметить, что контроль «временных интервалов» не способен выявить атаки, основанные на загрузке штатной ТОС в память нештатной рабочей станции, по следующим причинам[6]:

– использование нарушителем средств виртуализации с целью загрузки штатной ТОС в память виртуальной машины ($T_1 = T_{1\text{норм}}$ и т.д.) для создания «снимка» (snapshot) памяти;

– использование нарушителем электронных устройств негласного получения и передачи информации, внедренных в аппаратные компоненты нештатной станции и не влияющих на процесс загрузки ТОС ($T_1 = T_{1\text{норм}}$ и т.д.).

Наиболее распространенным на текущий момент времени техническим методом выявления и предотвращения атак такого типа является контроль целостности образа ТОС криптографическими методами с использованием программно-аппаратных средств защиты информации [6]. Однако в случае территориально нераспределенных автоматизированных систем такой подход является избыточным. Тогда одним из вариантов нейтрализации представленного типа атак может стать совокупность использова-

ния метода «временных интервалов» и контроля целостности линий связи и мест их подключения к рабочей станции вне зависимости от состояния станции в любой момент времени: выключена, включена, отсутствует электропитание.

СПИСОК ЛИТЕРАТУРЫ:

1. Тищенко Е.Н., Буцик К.А., Деревяшко В.В. Модель доверенной сетевой загрузки тонкого клиента с нейтрализацией возможностей внутреннего нарушителя // Известия ЮФУ. Технические науки. 2015. № 5 (166).
2. Брукс Ч. Дж. CompTIAA+. Устройство, настройка, обслуживание и ремонт ПК, 3 изд. Санкт-Петербург: БВХ-Петербург, 2010. С. 1232.
3. Хамбл Дж., Фарли Д. Непрерывное развертывание ПО: автоматизация процессов сборки, тестирования и внедрения новых версий программ / Пер. с англ. М.: ООО «И.Д. Вильямс». 2011.
4. Wall D. S. ENEMIES WITHIN: Redefining the insider threat in organizational security policy // Security Journal. Vol. 26(2). 2013. Pp. 107–124.
5. Wojtczuk R., Kallenberg C. Attacks on UEFI security, inspired by Darth Venamis's misery and Speed Racer // CanSecWest. Vancouver. 2015.
6. Чугринов А. В. Доверенные сеансы связи и средства их обеспечения // Информационная безопасность. 2010. № 4. С. 54–55.

REFERENCES:

1. TishenkoE.N., ButsikK. A., DerevyashkoV. V. Model doverennoy setevoy zagruzki tonkogo klienta s neitralizaciey vozmojnostey vnutrennego narushitelja // IzvestiyaSFedU. 2015. № 5 (166).
2. Charles J. Brooks. CompTIA A+ (Exams 220-602, 220-603, 220-604). St. Petersburg: BVH-Peterburg. 2010. P. 1232.
3. Jez Humble, David Farley. Continuous Delivery. – Addison-Wesley Publishing Company, Inc. 2011.
4. Wall D. S. ENEMIES WITHIN: Redefining the insider threat in organizational security policy // Security Journal. Vol. 26(2). 2013. Pp. 107–124.
5. Wojtczuk R., Kallenberg C. Attacks on UEFI security, inspired by Darth Venamis's misery and Speed Racer // CanSecWest. Vancouver. 2015.
6. Chugrinov A. V. Doverennie seansi svjazi I sredstva ih obespechenija // Information Security. 2010. № 4. Pp. 54–55.