М. И. Фроимсон, А. М. Рапетов, Н. В. Сычев

SMS-ДЕЗОРИЕНТАЦИЯ ПОЛЬЗОВАТЕЛЕЙ МОБИЛЬНЫХ ТЕЛЕФОНОВ

В настоящее время мобильное мошенничество набирает обороты и приносит все больше и больше прибыли злоумышленникам, а развитие технологий приводит к появлению новых угроз. Распространенным в последние годы стало использование поддельных SMS-сообщений, которые приходят с известных атакуемым абонентам номеров или с подписями от сотовых операторов. На рынке при этом не существует программных средств, которые защищали бы мобильные телефоны от данного типа атак.

Механизм подделки SMS-сообщений может использоваться для того, чтобы передать на мобильный телефон сообщение с искаженным номером отправителя. Такой прием используется мошенниками для того, чтобы ввести атакуемого в заблуждение и завладеть ценной информацией, получив ответ на знакомый ему номер, а также использовать переданную в SMS-сообщении информацию в целях дальнейшей атаки [1].

Рассмотрим основные нюансы атаки мошенников с использованием SMS-дезориентации.

В качестве отправителя SMS злоумышленник может указать любой номер телефона или символьное имя. Если будет указан номер, то аппарат атакуемого произведет поиск в контактной книге и отобразит на экране соответствующее номеру имя так, как оно записано в телефоне. Сообщение скорее всего не вызовет подозрения со стороны атакуемого.

Подделка имени отправителя SMS-сообщения позволяет не только вводить людей в заблуждение, но и проводить такие действия, как управление различной электроникой и сервисами. Например, уязвимые банковские сервисы — угроза для счета пользователя, а управляемый со стороны «умный дом» — угроза для его жизни.

Для атаки используется механизм запросов к коммерческим SMS-шлюзам. Дело в том, что, отправляя сообщение через SMS-шлюзы, можно параметрически задавать телефон отправителя.

Атаку злоумышленник может осуществить и более простым способом, указывая не телефонный номер, а текстовую строку. В таком случае поступившее SMS-сообщение может быть, например, от «Васи» или «Марины Алексеевны». Если в записной книжке телефона есть контакт с именем «Вася», то можно предположить, что атакуемый абонент поверит в подлинность сообщения. В этом случае, заметим, злоумышленнику даже нет необходимости знать телефон указанного человека. Практически безотказно действует в большинстве ситуаций простая подпись «Мама».

Для осуществления нападения злоумышленник использует вредоносный скрипт, работающий с SMS-шлюзом, предварительно зарегистрировав свой профиль и учетную запись на сервере шлюза [2].

Рассмотрим текст подобного скрипта на языке PHP.

```
<?
$user="user";
$password="password";
$api_id="xxx";
$url="http//some-sms-gate.com";
$text=urlencode("Текст сообщения");
$to="7903xxxxxxx";
$from="7926xxxxxxx";
```

```
$url="$url/http/auth?user=$user&password=$password&api_id=$api_id";
$ret= file ($url);
$sess=split(":", $ret[0]);
if ($sess[0]=="OK")
{
$sess_id=trim($sess[1]);
$url="$url/http/sendmsg?session_id=$sess_id&to=$to&text=$text&from=$from";
$ret=file($url);
$send=split(":",$ret[0]);
if ($send[0] == "ID") echo "success <br> message ID: ".$send[1];
else echo "send message failed";
}
?>
```

В начале скрипта происходит определение основных переменных user, password и api_id, т. е. логина, пароля и идентификатора пользователя, на SMS-шлюзе.

Текст сообщения записывается в переменную $text и должен иметь такое содержание, которое побудит атакуемого абонента к ожидаемым действиям. Телефонный номер жертвы записывается в переменную $to. Именно со счета этого абонента будут сниматься деньги.

В переменной $from указывается номер, который должен ввести в заблуждение пользователя мобильной связи. В это поле может быть помещен как номер телефона, так и текстовая строка. К SMS-шлюзу осуществляется запрос на начало сессии по отправке SMS-сообщений с помощью PHP-функции file. Ответ SMS-шлюза тут же проверяется на то, разрешена ли передача сообщения.

При наличии достаточных средств на счету, связанном с указанной учетной записью, SMS-сообщение передается.

Отправка сообщения выполняется с помощью функции file. Если она произведена успешно, то на экран выводится строчка: «Success. Message was sent».

В рассмотренном примере для работы с SMS-шлюзом использовался HTTP-протокол, который, с точки зрения злоумышленника, не является самым эффективным. Во-первых, большинство шлюзов имеют весьма ограниченный набор функций при работе по HTTP-протоколу, и далеко не все из них предоставляют возможность менять имя отравителя. Во-вторых, многие SMS-шлюзы принципиально не реализуют взаимодействие по HTTP, так как участились случаи жалоб на то, что к такому механизму отправки сообщений могут получить доступ все желающие, в том числе и злоумышленники.

Все большее количество SMS-шлюзов переходит на работу по SMPP.

SMPP (Short Message Peer to Peer) — протокол взаимодействия клиента и SMS-шлюза [3]. SMPP — гораздо более сложный протокол, чем HTTP, но при этом производительность его гораздо выше. Это объясняется тем, что он является бинарным и используется в режиме постоянного подключения, в то время как при работе с HTTP клиент устанавливает соединение, отправляет запрос, получает ответ сервера, а затем соединение закрывается. Постоянное подключение позволяет значительно повысить скорость передачи в случае отправки большого количества сообщений.

SMPP-протокол обладает достаточно мощными возможностями. Поэтому более внимательно рассмотрим функции, которые обеспечивают создание клиентского приложения и выполняют простую отправку SMS. Уязвимость именно этой части протокола часто используется злоумышленником.

Чтобы отправить сообщение по SMPP через SMS-шлюз, необходимо:
- подключиться к SMS-шлюзу;
- отправить серверу сообщение BIND_TRANSMITTER, указывающее на запрос со стороны клиента с целью создания постоянного соединения с SMS-шлюзом;
- дождаться от сервера ответа BIND_TRANSMITTER_RESP, который указывает на то, что запрос о создании соединения принят или отвергнут;
- отправить сообщение SUBMIT_SM, которое отвечает за отправку SMS-сообщения и содержит его текст;
- дождаться от сервера сигнала SUBMIT_SM_RESP;
- разорвать соединение, отправив сообщение UNBIND.

Рассмотрим более подробно каждый этап работы по SMPP-протоколу, чтобы выявить уязвимости, которые дают возможность злоумышленникам совершать атаку.

Подключение к SMPP-серверу.

SMPP-сервер — это часть SMS-шлюза, которая отвечает за работу с клиентскими приложениями по SMPP-протоколу.

На данном этапе пользователю достаточно иметь IP-адрес SMPP-сервера и номер порта, к которому нужно подключиться. Необходимо помнить о том, что многие SMPP-сервера не позволят соединения без процедур идентификации и аутентификации. Поэтому для подтверждения прав пользователя понадобятся логин и пароль. Помимо этого, как правило, все SMPP-сервера защищены межсетевыми экранами, и для каждого конкретного подключения IP-адрес клиента должен быть разрешен в системе сетевой защиты SMPP-сервера.

Такие меры позволяют SMS-шлюзам снизить вероятность атаки, а также отследить недобросовестного обладателя учетной записи на сервере.

Отправка BIND_TRANSMITTER.

Необходимо отметить, что работа по протоколу SMPP состоит в обмене пакетами данных между клиентом и сервером в обоих направлениях. Все сообщения, которыми они обмениваются, имеют стандартизованные названия, например: BIND_TRANSMITTER, BIND_TRANSMITTER_RESP и т. д. Каждый пакет состоит из нескольких частей — заголовка и непосредственно тела сообщения.

BIND_TRANSMITTER необходимо отправить для того, чтобы открыть сессию. Если попытаться сразу передать SUBMIT_SM, то SMPP-сервер сообщит об ошибке.

Сессия — это некое состояние, после установления которого можно посылать и принимать SMS-сообщения. При открытии сессии происходит авторизация клиента, проверка его баланса и возможности передачи сообщений.

Вообще говоря, настоящий клиент должен сделать и закрытие сессии, отправив сообщение UNBIND и дождавшись сообщения UNBIND_RESP.

Следует заметить, что злоумышленник не делает этого, поскольку задача атакующего только отправить SMS, затратив при этом минимум усилий.

Ожидание PDU BIND_TRANSMITTER_RESP.

До того, как придет BIND_TRANSMITTER_RESP, сессию нельзя считать открытой, а потому никакие другие сообщения отправлять не следует. Получив BIND_TRANSMITTER_RESP, нужно убедиться в том, что значение поля «статус» в заголовке равно нулю. Это означает отсутствие ошибок при выполнении команды.

Отправка SMS с помощью SUBMIT_SM.

Кульминацией работы по SMPP-протоколу является отправка SMS. В случае с рассматриваемым протоколом речь идет о передаче сообщения, которое включает в себя адрес

получателя, адрес отправителя, текст и множество других параметров. Как правило, служебная информация содержится в заголовке (время отправки, кодировка и т. д.), а само сообщение следует сразу за ним.

Ожидание сообщения SUBMIT_SM_RESP.

Этим сообщением сервер подтверждает факт принятия SMS в обработку.

Разрыв соединения.

Перед разрывом самого соединения следует закрыть сессию, отправив сообщение UNBIND и дождавшись сообщения UNBIND_RESP.

Теперь перейдем к рассмотрению действий злоумышленника при возможной реализации данной атаки. Приведем пример, написанный на языке программирования PHP.

```
<?
require_once ('smppclass.php');
$smpphost = "203.199.142.41";
$smppport = 2345;
$systemid = "user";
$password = "pswd";
$system_type = "Rdsd";
$from = "79035050210";
$smpp = new SMPPClass();
$smpp->SetSender($from);
$smpp->Start($smpphost, $smppport, $systemid, $password, $system_type);
$smpp->TestLink();
$smpp->Send("0123456789", "Message");
$smpp->End();
?>
```

В первой строке подключается специальный класс PHP для работы с SMPP-протоколом. Отметим, что подобные объекты реализованы практически для всех языков программирования, чтобы пользователь мог, не вникая в тонкости работы бинарного протокола, реализовать свое приложение.

Далее идет блок, где определяются основные переменные, которые позже используются в программе. В переменных $smpphost и $smppport определены IP-адрес и порт SMPP-сервера соответственно. Далее в переменных $systemid, $password и $system_type определяются необходимые для авторизации на сервере данные: логин и пароль, а также идентификатор пользователя на SMPP сервере.

В переменной $from определяется ключевая с точки зрения атаки информация — номер отправителя, который может быть любым.

Далее идет непосредственная реализация кода программы. Создается новый класс SMPP, который призван облегчить работу с SMPP-протоколом.

Затем злоумышленник устанавливает номер отправителя с помощью команды класса Set-Sender.

После этого осуществляются подключение к SMPP-серверу и авторизация.

Эта сложная операция производится всего лишь одной строчкой кода с использованием объявленной в классе функции Start, которая осуществляет как подключение к серверу, так и отправку сообщения BIND_TRANSMITTER, а также получение сообщения BIND_TRANS-MITTER_RESP.

Обмен пакетами SUBMIT_SM и SUBMIT_SM_RESP также выполняется с помощью функции класса Send, которой передаются два параметра: номер получателя и текст сообщения. И наконец, окончание соединения оформляется с помощью функции End класса SMPPClass().

Таким образом, атака с помощью SMS-дезориентации может эффективно выполняться и на серверах, не поддерживающих взаимодействие с клиентом по протоколу HTTP.

Данной проблеме не уделяется должного внимания со стороны сотовых операторов. Дело в том, что обнаружить подобную атаку и закрыть подозрительный SMS-шлюз достаточно легко, и считается, что защита в данном случае осуществляется со стороны SMS-шлюза, так как для него подобные атаки просто невыгодны. Тем не менее существует целый ряд таких служб, расположенных преимущественно в развивающихся странах, в том числе и в странах Ближнего Востока, позволяющих регистрацию пользователей без предоставления паспортных данных и без какого-либо контроля их деятельности, что даже в случае обнаружения факта проведения атаки делает невозможным привлечение злоумышленника к ответственности. Также считается, что осуществлять подобную атаку невыгодно, так как для получения прибыли необходимо отправить огромное количество сообщений, что привлечет внимание сотового оператора, и шлюз будет быстро заблокирован. К сожалению, существуют ситуации, когда пользователи мобильных телефонов особенно подвержены угрозе мошенничества, и за время, необходимое для блокировки шлюза, злоумышленник сможет отправить достаточное количество сообщений. К таким ситуациям относятся террористические акты. Во время взрывов большинство людей в ответ на сообщение с даже незнакомого номера с просьбой срочно перевести деньги на указанный абонентский счет сделают это почти не задумываясь, беспокоясь о своих близких. Таким образом, используя потерю бдительности людьми во время чрезвычайных ситуаций, злоумышленники фактически за один день могут успеть собрать, по приблизительным оценкам, около 2 млн долларов. С учетом этого факта можно отмести довод о невыгодности подобных атак.

Более того, становится очевидной связь проведения террористических актов, вызывающих панику среди населения, с получением финансовой выгоды подобным путем. Поэтому недостаточная степень защиты от атак с использованием SMS-шлюзов может не только привести к потере средств абонентами, но и представлять угрозу для многих человеческих жизней, что является гораздо более страшным последствием широкого распространения нелицензируемых SMS-шлюзов.

В качестве защиты от подобных атак на уровне мобильного телефона можно предложить встроенную в операционную систему, установленную на нем систему аутентификации SMS-центров, с которых приходят принимаемые пользователем сообщения. Определение подлинности и вредоносности SMS-сообщений должно происходить на основе анализа SMS/MMS PDU (Protocol Data Unit) и DEF-кодов номера отправителя и SMS-центра. К сожалению, в настоящее время подобные механизмы отсутствуют в существующих мобильных платформах, чем обосновывается необходимость разработки средств защиты от рассмотренных атак.

СПИСОК ЛИТЕРАТУРЫ:

1. *Ле-Бодик Г.* Мобильные сообщения: службы и технологии SMS, EMS и MMS / Пер. с англ. М.: КУДИЦ-ОБРАЗ, 2005.
2. *Зуйков А. В., Михайлов Д. М., Стариковский А. В., Фроимсон М. И.* Уязвимость системы коммерческих SMS-шлюзов в инфраструктуре GSM-сетей // Сборник материалов II Ежегодной Всероссийской научно-практической конференции с международным участием. Новосибирск: СИБПРИНТ, 2010. С. 321–326.
3. *Henry-Labordere A., Jonack V.* SMS and MMS interworking in mobile networks. Artech House Inc., Boston 2004.