Марина Гвозденко,
Ярослав Чобу
(Харків)

HONEYPOT ЯК ЗАСІБ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

Honeypot – це мережевий ресурс, мета існування якого – це «приманка» зломщика, його ідентифікація і відстеження дій, які він вживає для злому вузла. Це одна з технологій, яка покликана забезпечити мережеву безпеку. Засоби honeypot відрізняються від класичних засобів забезпечення безпеки тим, що вони не покликані вирішувати яку-небудь конкретну задачу [1, с. 17]. Навпаки, honeypot – гнучке засіб, який може бути застосоване в різних ситуаціях. Назва походить від англійського повір'я, згідно з яким якщо залишити горщик з медом (вразливий вузол), то на нього обов'язково злетяться бджоли (хакери). Протистояти honeypot'ам досить складно, але цілком реально. Мета цієї роботи – дати загальне уявлення про структурі honeypot'ов і про деякі способи протистояння їм.

Технологія honeypot'а надає аналітикам такі переваги: збір інформації про зломщиків, невимогливість до системних ресурсів, простота управління і наочність необхідності використання. Зазвичай, системи IDS (виявлення вторгнень) реєструють десятки, а то і сотні мегабайт інформації за день. В цій горі журналів не так легко відшукати потрібні відомості, бо більшість запитів будуть легальними запитами від користувачів. Honeypot журналює незрівнянно менші обсяги інформації, які на 100% містять інформацію, необхідну для аналізу (якщо система правильно налаштована). З цього випливає, що honeypot невимогливий до ресурсів в силу свого призначення, йому не потрібні оновлення або постійна технічна підтримка, досить якось правильно налаштувати й очікувати [2]. Також, honeypot'и є наочним прикладом того, що в них не даремно вкладені гроші. Якщо фірма вкладає гроші в IDS і подібні засоби захисту, а потім ці кошти надійно захищають мережу, то може скластися враження, що це була марна трата грошей, так як мережу ніхто не зламує.

Honeypot в цьому випадку буде хорошим доказом того, що мережа все-таки зламують і гроші вкладені не дарма.

Залежно від цілей, які переслідує honeypot, він може мати різні варіанти конфігурації, починаючи від рівня програмного забезпечення, яке не вимагає особливого налаштування, закінчуючи складними апаратними комплексами. Залежно від рівня складності honetpot'a і його можливостей, їх можна класифікувати на 3 групи: honeypot'и слабкого, середнього і сильного рівня взаємодії. Розберемося з кожним з них по порядку.

Honeypot'и слабкого рівня взаємодії прості у використанні і вельми надійні. Вони імітують тільки частину сервісів, і зломщик буде обмежений у взаємодії з ними. Приміром, вони можуть імітувати систему UNIX із запущеним сервісом telnet. При підключенні до такого вузлу зломщик отримає запит login і намагатиметься підібрати паролі до системи. Або, припустимо, FTP–сервер з анонімною обліковим записом і нібито файлом з паролями (номерами кредитних карт та ін.). Природно, що будь-яка спроба доступу до цього файлу означатиме спробу злому. Система збереже в журналах час, коли відбулася спроба злому, IP адресу і порт зловмисника, а також порт, до якого він спробував отримати доступ. Завдання таких програмних honeypot'ов – мінімальний рівень протоколювання. Такі системи розраховані на самих початківців зломщиків. Ризик при використанні honeypot'ов 1-го рівня мінімальний, але він є. Це пов'язано з тим, що сам цей програмний комплекс теж програма, отже, може бути уразливий. Якщо його вдасться обійти, то зломщик отримає доступ до решти вузлам мережі. Сила цих найпростіших honeypot'ов в тому, що вони самі по собі прості. Як відомо, чим простіше, тим надійніше, тому ці програми мінімізують ризик, пов'язаний з можливим зломом самого honeypot'в і подальшим зломом системи.

Honeypot'и середнього рівня взаємодії надають більше можливостей з реконструкції дій зломщика, більш складні, отже, більш уразливі. Наприклад, така система може моделювати складніші веб-сервери, які зможуть реагувати на нестандартні команди і будуть мати більш складну систему журнулювання.

В ОС UNIX можна використовувати можливості команди chroot, а в Windows – систему віртуальних машин VMWare. Таким чином, розширитися оточення зломщика (тобто, він зможе взаємодіяти не лише з «підробленими» сервісами, але і з «підробленою» ОС), і це дасть більше можливостей для протоколювання. Але такий підхід також створить більше проблем. По-перше, дане рішення досить складне, таким чином, на стадії роботи або конфігурування можуть виявитися помилки, які потім приведуть до більшої вразливості системи. По-друге, надати віртуальному оточенню функціональність реальної системи (хоча б зімітувати) – трудомістке завдання. Чим більше функціональності і реалістичності надається віртуальному оточенню, тим легше для зловмисника обійти дане оточення і отримати контроль над реальною операційною системою. До того ж, зловмисний код може вийти з-під контролю віртуальної машини, якщо він спеціально спроектований під це.

Honeypot'и сильного рівня взаємодії надають максимум інформації про зловмисника і максимально складні і небезпечні. Вони надають зловмисникові доступ до реальної системи, яка нічого не робить і не пов'язана з іншим системами. Структура такого honeypot'a найчастіше така: вузол-приманка, мережевий сенсор і накопичувач інформації. Такий вузол може бути розташований в мережі за брандмауером, і тоді власне контроль лягати на брандмауер. Якщо вузол приманка буде неправильно налаштований або виникнуть ще якісь непередбачені речі, то зломщик зможе отримати доступ до мережі. Одним з недоліків такого рішення може бути складність його реалізації і відносна дорожнеча підтримки [1, с. 79].

А тепер можна розглянути питання про недоліки (не рахуючи тих, які вже були названі) різних способів реалізації honeypot'a. Сама ідея honeypot'a припускає наявність ресурсу, який буде привертати увагу потенційних зломщиків. Легальних звернень до нього бути не повинно, і всяке звернення до нього повинно викликати підозру у адміністратора. Однак ізольованість вузла від інших вузлів мережі вже повинна викликати підозру і гостре бажання покинути ресурс пошвидше. Якщо «приманка» не має ніякого трафіку крім

хакерського (це теж можна визначити), то потрібно йти. Але навіть якщо вузол взаємодіє з деякими іншими, це може бути як просто віртуальна локальна мережа, що створює «видимість» взаємодій, або інші вузли–honeypot'и, які налаштовані на різний рівень «складності злому»: від найпростіших, які містять дистрибутиви по замовчуванням, до найбільш захищених, які націлені на виявлення невідомих атак. Якщо в мережі видний явно уразливий вузол (та особливо з конфігурацією за умовчанням; діри таких конфігурацій добре відомі), то це або брак знань адміністратора, або пастка. Наслідки від другого можуть бути набагато більш плачевними, ніж якась вигода в разі першого. Honeypot 1-го рівня може бути впізнаний, якщо направити йому нестандартну команду, або якщо це популярна система, яка містить помилки. У такому випадку вона сама може стати об'єктом атаки [3]. Як вже говорилося, зловмисний код може вийти з–під контролю віртуальної машини (VMWare не ідеальна і теж містить помилки), тоді зломщик отримає контроль над батьківською системою (стосовно до honeypot'ів 2-го рівня). Якщо зломщик має справу з honetpot'ом 3 рівня взаємодії, то тут можуть бути різні варіанти. Можна користуватися вразливостями конфігурації самого honeypot'а. Наприклад, невірно налаштований мережевий сенсор (який веде базу даних – «серце» всієї системи) може або видавати помилкові попередження після кожного сканування портів, або не реагувати на злегка видозмінені атаки.

Взагалі ж, при зломі вельми серйозних ресурсів, таких як державні сайти, банки тощо зломщики можуть використовувати різноманітні стратегії. IP-адреса зломщика – це ще не показник. Це може бути IP-адреса proxy-сервера. В цьому випадку залишається лише покладатися на те, що proxy-сервер веде журнал роботи. Тому деякі зломщики використовують ланцюжок з декількох зламаних комп'ютерів, а в мережу виходять через GPRS / EDGE-модем в мобільному телефоні (який можна швидко знищити в разі необхідності), від'їхавши подалі від свого місця проживання. До того ж, якщо машина зломщика вразлива, то honeypot може закинути на його комп'ютер вірус, або зібрати інформацію, наприклад, через cookie.

Вибравши жертву, зломщик повинен максимально вивчити топологію її мережі. Він повинен переконатися, що вузол обслуговує зовнішній трафік, що конфігурація відмінна від конфігурації за замовчуванням, що вузол використовується іншими учасниками мережі і т. ін. Потім він може кілька днів нагнітати обстановку, скануючи порти і засилаючи рядка, що імітують переповнення буфера. Також він може атакувати сам honeypot для виведення його з ладу (DDoS, SYN, ECHO-death та інші атаки, які дозволяють приховати IP зломщика). Можна також переповнити накопичувач інформації та система деякий час не зможе зареєструвати злом.

Загалом, honeypot'и зроблені людьми, а тому теж уразливі. До того ж, вони є лише частиною «оборонної системи». Покласти всю надію на honeypot'и нерозумно. Поки користувачі використовують паролі типу password123 і працюють під обліковими записами з привілеями адміністратора, а системні адміністратори, один раз встановивши і налаштувавши комп'ютери, не приділяють системам належної уваги, то навіть сама остання IDS і найсучасніші антивіруси з брандмауерами не зможуть захистити мережу. Потрібно використовувати комплексний підхід, в якому honeypot'ам відведена власна ніша. Вони повинні працювати за добре налаштованим брандмауером і бути ізольованими від мережі. З користувачами потрібно проводити бесіди про те, які паролі краще вибирати і які вимоги до них повинні бути. Системний адміністратор повинен підвищувати свій рівень кваліфікації, регулярно оновлювати систему і стежити за її станом. А керівництво не повинно економити на безпеці даних, інакше це потім дорожче вийде.

ДЖЕРЕЛА ТА ЛІТЕРАТУРА:

1. Джесси Рассел. Honeypot (информационная безопасность) / Джесси Рассел. – М.: Книга по Требованию, 2013. – 106 с.
2. Електронний ресурс. – Режим доступу: http://www.hackzone.ru /articles/ view/id/8993/
3. Електронний ресурс. – Режим доступу: http://www.at–soft.com.ua