

Цель работы

Показать этапы реализации атак и контрмер в сценарии компрометации научно-технической информации предприятия, определить уязвимости, продемонстрировать доказательства успешной эксплуатации и предложить практические способы устранения.

Теоретическое введение

Сценарий №5

Защита научно-технической информации предприятия Внутренняя служба безопасности не смогла обнаружить в новом сотрудники специально подготовленного агента, который устроился в компанию для получения сведений, касающихся разработки новых насосных станций. Внутренний нарушитель проводит ряд успешных атак как на внутренних сотрудников компании, так и на сервера ЦОД. В результате он смог подключиться к внутренней базе данных и получить значения технических параметров работы новых насосных станций. Квалификация нарушителя высокая. Он умеет использовать инструментарий для проведения атак, а также знает техники постэксплуатации.

Уязвимость 1: Слабый пароль учётной записи dev1 — позволяет перебор словарём.

Последствие 1: Developer backdoor — успешная загрузка и исполнение вредоносного файла, установка задачи в планировщике и Reverse Shell.

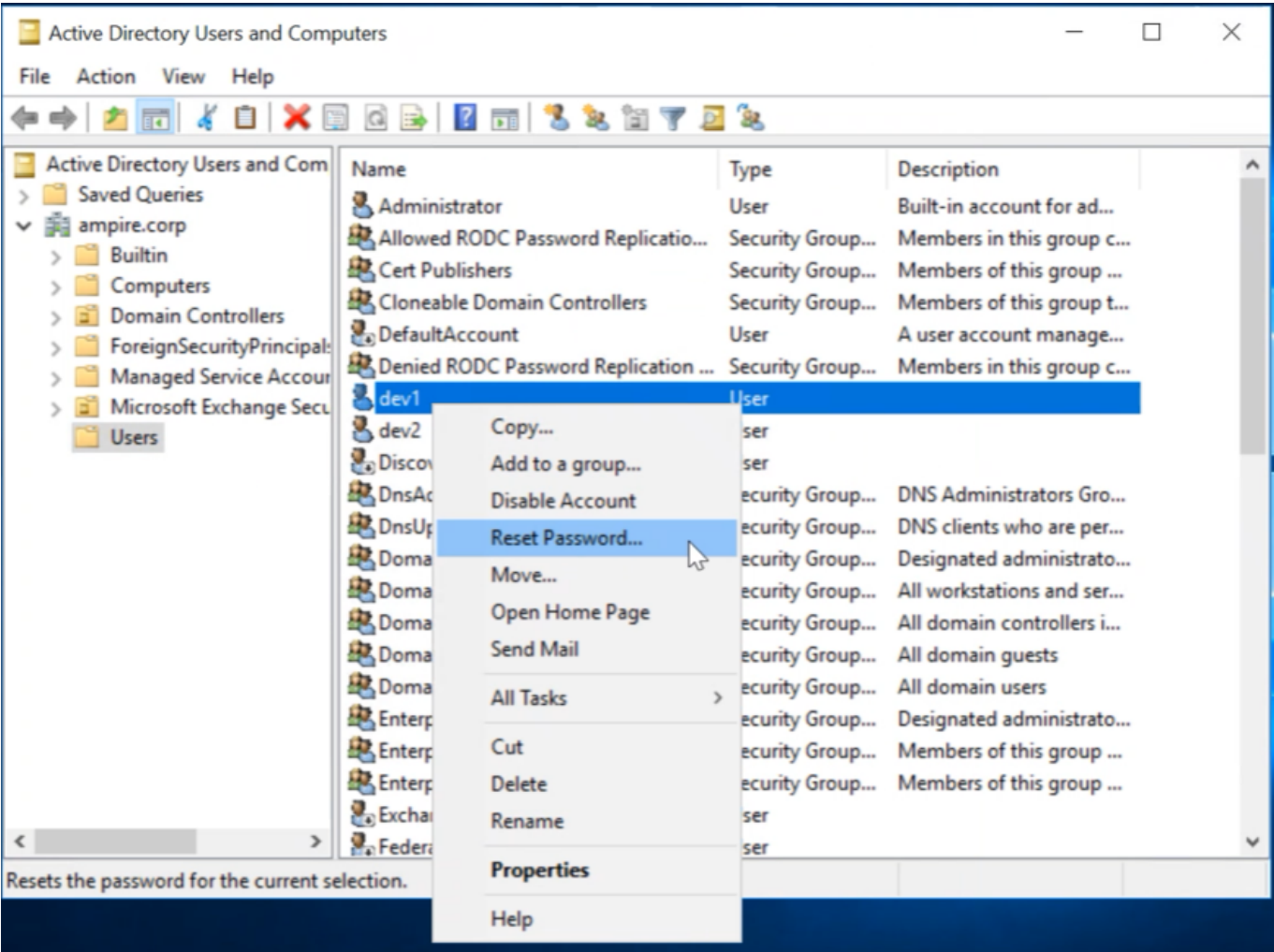
Уязвимость 2: Stored XSS в Redmine (CVE-2019-17427) — позволяет исполнить вредоносный код через textile-разметку wiki.

Последствие 2: Создание административного пользователя Redmine (Redmine User) и дальнейшее расширение привилегий.

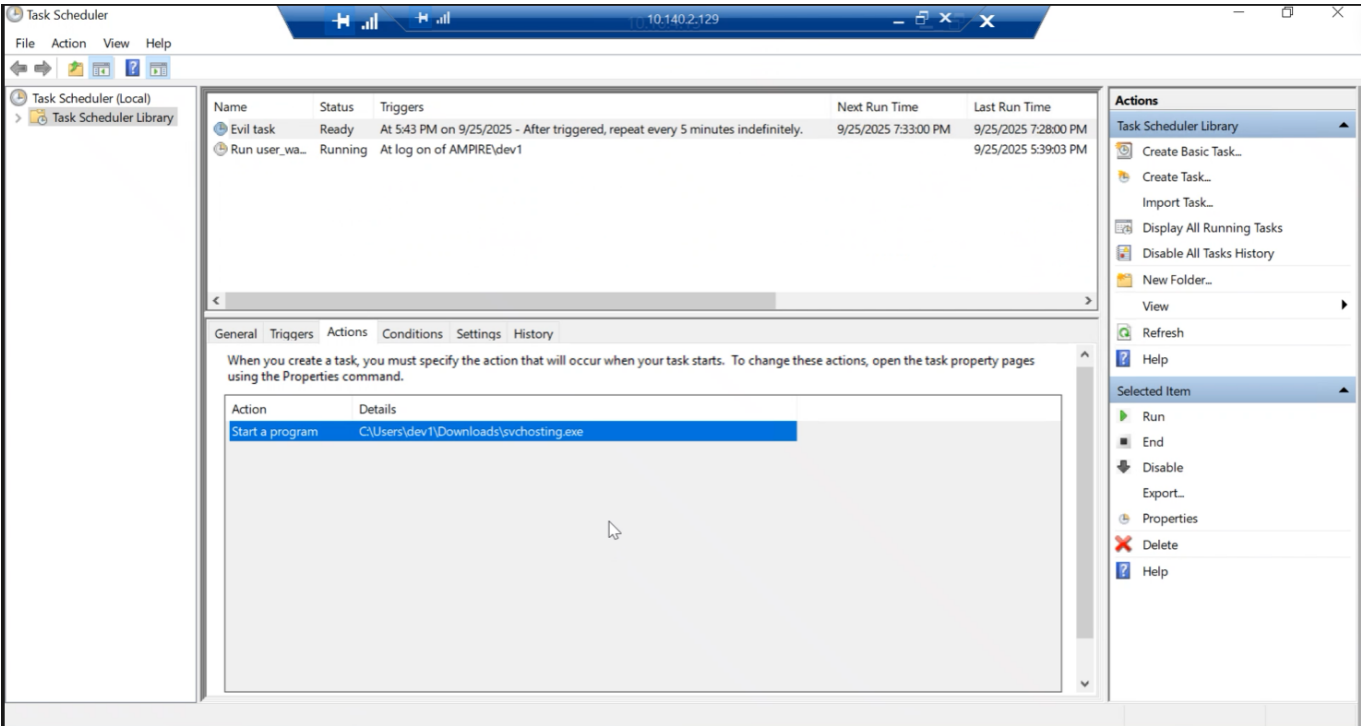
Уязвимость 3: Blind SQL (CVE-2019-18890) — позволяет извлечь защищённые данные через по-символьные запросы с измерением времени ответа.

Выполнение лабораторной работы

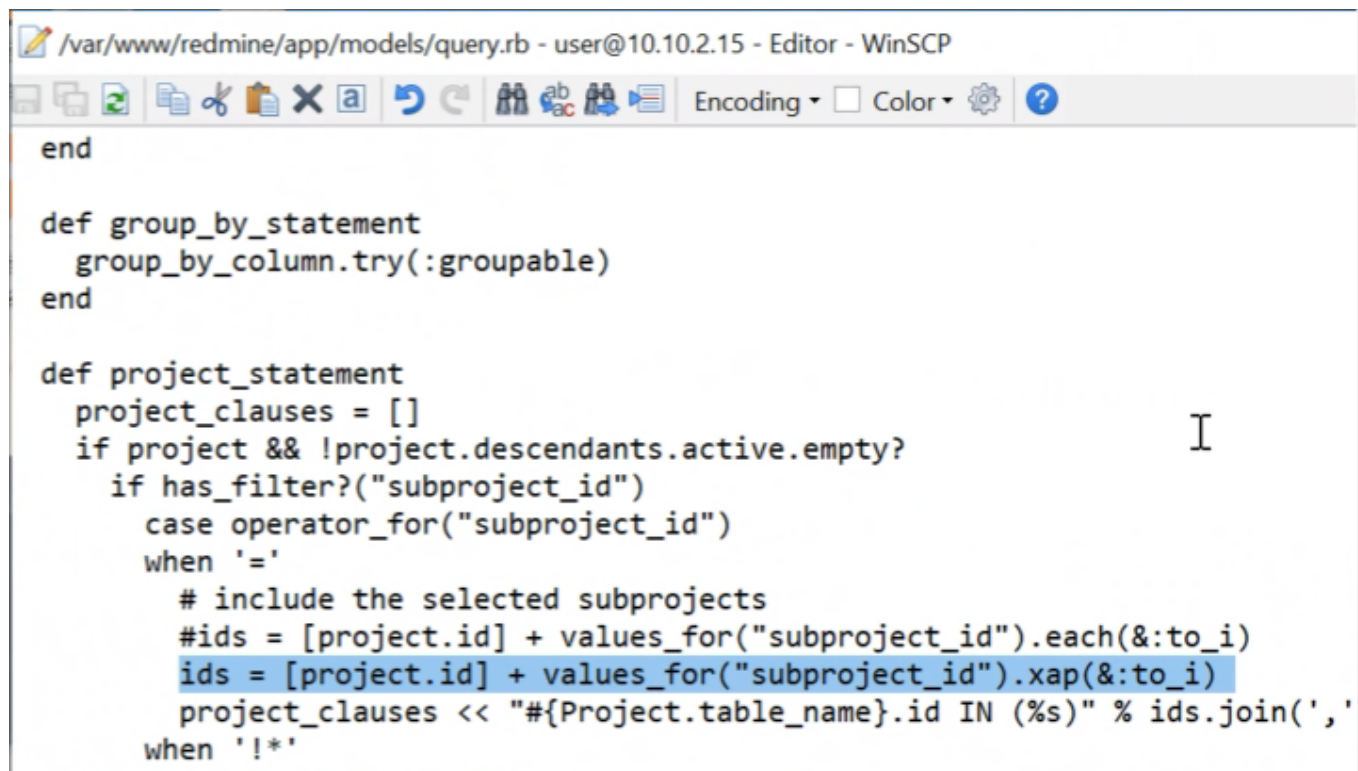
Для устранения уязвимости 1 мы сменили пароль у dev 1 на более сложный (рис. @fig:001).



Злоумышленник с помощью уязвимости добавил вредоносную нагрузку, которая создаёт задачу в Планировщике задач Windows для автозапуска evil task (рис. @fig:002). Для устранения этого последствия мы удалили задачу.



Уязвимость 2:



```
/var/www/redmine/app/models/query.rb - user@10.10.2.15 - Editor - WinSCP

end

def group_by_statement
  group_by_column.try(:groupable)
end

def project_statement
  project_clauses = []
  if project && !project.descendants.active.empty?
    if has_filter?("subproject_id")
      case operator_for("subproject_id")
      when '='
        # include the selected subprojects
        #ids = [project.id] + values_for("subproject_id").each(&:to_i)
        ids = [project.id] + values_for("subproject_id").xap(&:to_i)
        project_clauses << "#{Project.table_name}.id IN (%s)" % ids.join(',')
      when '!*'
      end
    end
  end
end
```

Вывод

В ходе выполнения данной лабораторной работы был изучен сценарий атаки на систему защиты научно-технической информации предприятия и способы её нейтрализации. Было рассмотрено, как цепочка уязвимостей может привести к компрометации данных, и какие меры позволяют устранить последствия.

Список литературы{.unnumbered}
