

Network Safety

Scala

NetworkSafety

1. 网络攻击介绍
2. 网站攻击介绍
3. 其他信息介绍

网络攻击:

常见网络攻击介绍:

1.DOS:

拒绝服务器攻击:拒绝服务就是用超出被攻击目标处理能力的数据包消耗可用系统、宽带资源,致使网络服务瘫痪的一种攻击手段。

比如我们常用的Ping命令来测试一个网络ip是否连接。但是当同时用Ping的命令几百万台的时候了?服务器就吃不消了。

简单的ping可以直接屏蔽掉。但是真正的DOS攻击利用我们网络服务进行攻击。是很难屏蔽掉的。因为我们无法区分攻击者是否为普通人还是机器。

有一个非常简单和形象的列子。知道DOS攻击是什么样子吗?请参考春节时候的12306

简单的可以通过防火墙规则解决。针对业务的无解决办法。最多封ip

网络攻击

中间人攻击：

攻击者假冒身份来骗取信息。

A---B。依靠的是ip，但是ip是谁分配的。又怎么确定这个ip对应的是这个用户了。靠的是去中心话的一个路由表。

当一个人控制了你的网络。不让你的地址在路由上注册

A--B(假)--B。那么你的消息都要经过他跑一段时间。那么他是中间人。你和B的消息没有任何隐私。信息窥探

平常我们的信息没有加密。别人都能看的到。但是。我们现在可以加密。只有A---B能看到，别人就看不到了。但是这种加密有一个问题。怎么确认B是B。而不是假B了。就有个东西叫证书。证明你是谁。那么证书又怎么被确认的了。有个东西叫根证书。

XSS注入攻击

XSS注入往往是针对的业务逻辑的一种攻击方式。

攻击者利用与服务器的各种交互时的各种参数。资源。插入一部分js代码。这种js代码再次被解析的时候。浏览器会认为是服务器的js代码而主动执行。

这就是xss注入。

xss注入分为两种。存储型和触发型。存储型是把恶意代码作为内容直接上传到服务器。可以大规模触发。

而触发型需要用户主动去跳转。

<http://www.wooyun.org/bugs/wooyun-2010-023391>

活生生的示范。对于图片参数没有过滤。可以伪造成url+onLoad(js)文件。

SQL注入

<http://www.cnblogs.com/hkncd/archive/2012/03/31/2426274.html>

简单说是利用SQL参数的不过滤检查造成的。

攻击者同样伪造语句

比如一个简单的Login语句是 'select * from users where username=? and password=?'

那么攻击者不知道账号密码。那么他可以这样做。

账号乱输:密码输入” or “1=1

select * from user where name="friddle" and hashpwd="" or "1=1"

然后成功注入，继续可以做一些事情

全部依赖后面怎么注入。

解决办法:用ORM或者把参数进行一次过滤

乌云

不得不说的乌云：

乌云是国内做安全的网站，大家可以在上面看大神们怎么刷腾讯ali漏洞的

<http://www.wooyun.org>

不得不说的58同城：

因为上了招聘电视招了一个没有真才实学的做安全的毕业生。

然后活活的被国内做安全的刷爆。

刷爆。

道哥：阿里安全第一人

<http://taosay.net/>

Scala: 基于Jvm上的语言

动态语言这么火。Python/Ruby被吹上了天。

函数式语言这么火。Lisp, Erlang, Cojure第二春

下期在介绍其他语言。但是我们简单介绍Scala

Scala

为何而来？

写Java是一件累人的事情。

多核时代。Java处理并发好像不怎么给力？

函数式语言最近很火？

Jvm性能优异，不想放弃？
(Twitter ruby->java)

无数的库。说放弃是一件难事

有什么优点？

Scala代码量少。(程序员必要要求)

Scala是函数式编程。对并发天生靠近，而且有很多并发特性(Erlang鲜活的列子)

性能不差(人家可不是无类型语言哦)

Jvm性能优异。(毕竟这么多年了)

对Java的良好继承

JavaIDE的良好支持(eclipse說的就是你)

Scala: HelloWorld

```
object HelloWorld {  
  def main(args: Array[String]) {  
    println("Hello, world!")  
  }  
}
```

代码高效：

相比Java具有更高层及的抽象函数式语言的各种高级功能。

函数可以作为参数传递。

有LazyInit, yield, map, reduce等各种高级功能(其实我也不会)

具有更好的并发能力, 事件触发
省去很多并发检查代码

运行高效：

同样生成JVM byte code

动态语言最大的问题是因为变量动态。在运行的时候不停的检查类型。

但是Scala不是动态语言。但用了类型推导。

用起来跟动态语言差不多。但是效率上相差巨大

代码对比

java:

```
for(int a=0;a<10;a++)  
{  
    for(int b=0;b<10;b++)  
    {  
        System.out.println(a+b)  
    }  
}
```

scala:

```
for(a<-1 until 10;b<1- until 10)  
    print(a+b)
```

函数式编程

```
def makesum(a)=a // val makesum(a)=a
def dofunc(fun:Int=>Int;b:Int)={
    return fun(b)
}
def main(args: Array[String]) {
    print(dofunc(makesum, 1))
}
```

函数
作为
参数
可以
直接
传递

Concurrent

```
import akka.actor.{ Actor, ActorSystem, Props }
```

```
val system = ActorSystem()
```

简单的消息注册和服务

```
class EchoServer extends Actor {  
  def receive = {  
    case msg: String => println("echo " + msg)  
  }  
}
```

```
val echoServer = system.actorOf(Props[EchoServer])  
echoServer ! "hi"
```

Scala缺点

IDE们还不支持联合编译。(这是最大的问题)
古老的框架支持不太好。比如**AWT**

Scala前景

国外 Scala: LinkedIn, EDFT, Twitter, etc等开始采用Scala作为开发语言

国内 Scala: 淘宝部分站点开始用这个语言, 杭州19楼全面采用

Eclipse/NetBean/Intellig 都支持Scala开发。

作为Jvm上的动态语言。与Java的无缝连接。开发高效。同时又不存在效率问题。前景只会越来越大。

IDEA一点补充

对于Tomcat的调试。Ultimate可实现。内置Tomcat插件。不需安装

Keygen我有

对于Spring, maven等的插件支持良好。

Maven还可以自动查找maven包。下载源代码解析。

JRebel是一个插件可以实现classes动态加载。即只要编译完Class。自动Tomcat更新。破解和配置有点麻烦。但是非常实用。有想法的可以详细了解

下期预告：

语言大串烧

Python(friddle)

Ruby(陈江)

Golang(袁帅, 假如能来的话)

Php(iehong或者袁帅)

Thanks for Watching

运维

分布式和云技术

自动化工具/找找

嵌入式开发：

库介绍：

Hadoop

Java6,7,8特性

GC

网络工程