

# 服务安全与监控

**NSD SECURITY**

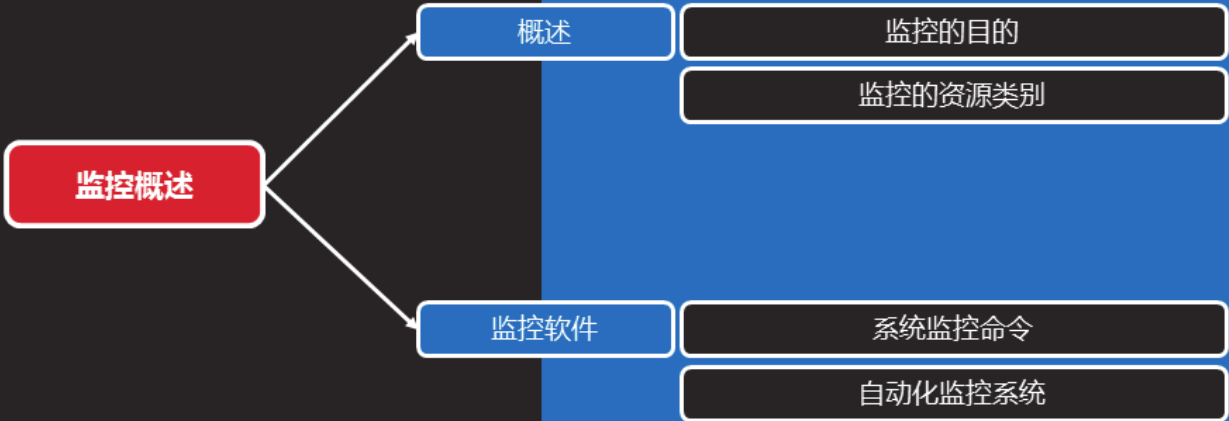
**DAY05**

# 内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	监控概述
	10:30 ~ 11:20	Zabbix基础
	11:30 ~ 12:00	
下午	14:00 ~ 14:50	Zabbix监控服务
	15:00 ~ 15:50	
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



## 监控概述



# 概述



## 监控的目的

- 报告系统运行状况
  - 每一部分必须同时监控
  - 内容包括吞吐量、反应时间、使用率等
- 提前发现问题
  - 进行服务器性能调整前，知道调整什么
  - 找出系统的瓶颈在什么地方



# 监控的资源类别

知识讲解

- 公开数据
  - Web、FTP、SSH、数据库等应用服务
  - TCP或UDP端口
- 私有数据
  - CPU、内存、磁盘、网卡流量等使用信息
  - 用户、进程等运行信息



# 监控软件

---

## 系统监控命令

知识讲解

- ps
- uptime
- free
- swapon -s
- df -h
- ifconfig
- netstat或ss
- ping
- traceroute
- iostat



## 自动化监控系统

知识讲解

- Cacti
  - 基于SNMP协议的监控软件，强大的绘图能力
- Nagios
  - 基于Agent监控，强大的状态检查与报警机制
  - 插件极多，自己写监控脚本潜入到Nagios非常方便
- Zabbix
  - 基于多种监控机制，支持分布式监控



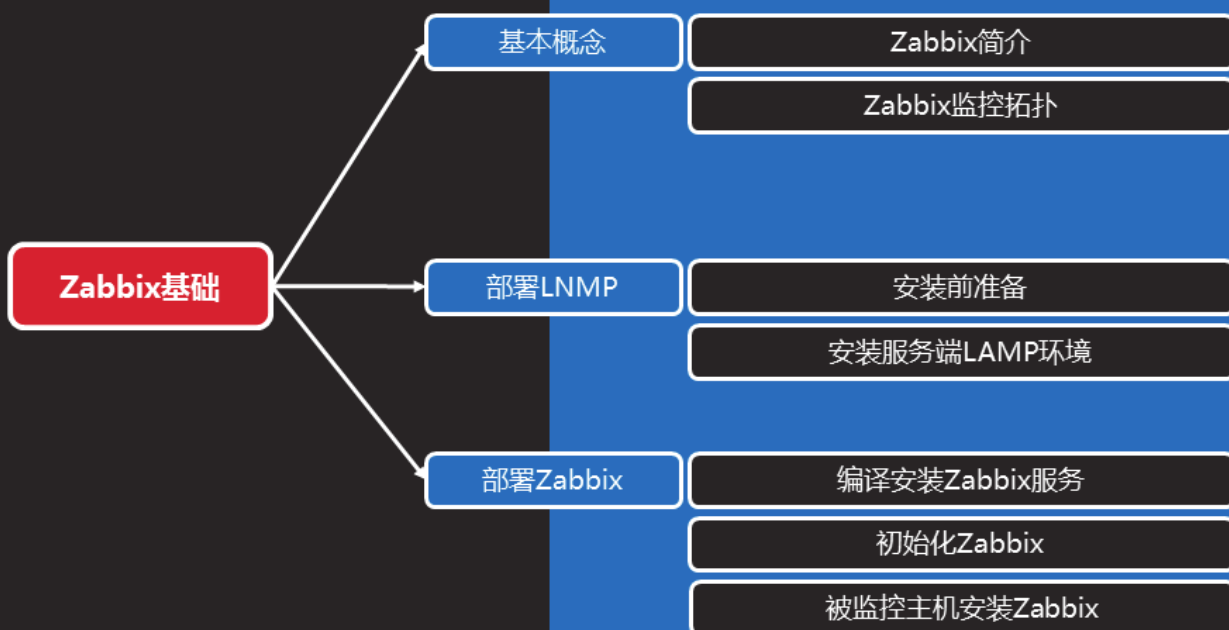
# 案例1：常用系统监控命令

课堂练习

- 使用系统命令查看系统性能参数
  - 查看内存信息
  - 查看交换分区信息
  - 查看磁盘信息
  - 查看CPU信息
  - 查看网卡信息
  - 查看端口信息
  - 查看网络连接信息



## Zabbix基础



# 基本概念



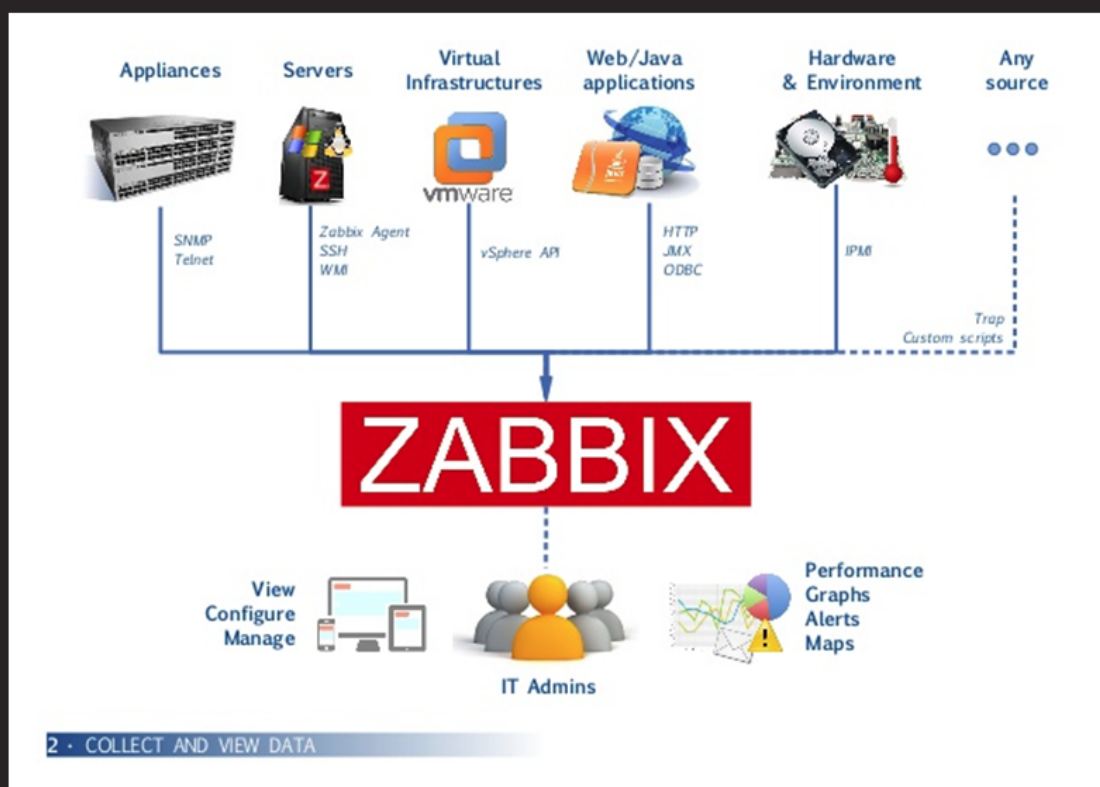
## Zabbix简介

- Zabbix是一个高度集成的监控解决方案
- 可以实现企业级的开源分布式监控
- Zabbix通过C/S模式采集监控数据
- Zabbix通过B/S模式实现Web管理



# Zabbix监控拓扑

知识讲解



## Zabbix监控拓扑（续1）

知识讲解

- 监控服务器
  - 监控服务器可以通过SNMP或Agent采集数据
  - 数据可以写入MySQL、Oracle等数据库中
  - 服务器使用LNMP实现web前端的管理
- 被监控主机
  - 被监控主机需要安装Agent
  - 常见的网络设备一般支持SNMP





# 部署LNMP

## 安装前准备

知识讲解

- 监控服务器
  - 设置主机名 ( zabbix server)
  - 设置IP地址 ( 192.168.2.5 )
  - 关闭防火墙、SELinux
- 监控客户端 ( 2.100和2.100 )
  - 主机web1 ( 192.168.2.100 )
  - 主机web2 ( 192.168.2.200 )
  - 关闭防火墙、SELinux



# 安装服务端LNMP环境

- 安装nginx及其依赖包

知识讲解

```
[root@zabbix server ~]# yum -y install gcc pcre-devel openssl-devel
[root@zabbix server ~]# tar -xf nginx-1.12.tar.gz
[root@zabbix server ~]# cd nginx-1.12
[root@zabbix server nginx-1.12]# ./configure --with-http_ssl_module
[root@zabbix server nginx-1.12]# make && make install
[root@zabbix server ~]# yum -y install php php-mysql \
> mariadb mariadb-devel mariadb-server
[root@zabbix server ~]# yum -y localinstall php-fpm-5.4.16-42.el7.x86_64.rpm
```



## 安装服务端LNMP环境（续1）

- 修改nginx配置

知识讲解

```
[root@zabbix server ~]# vim /usr/local/nginx/conf/nginx.conf
http{
    ... ..
    fastcgi_buffers 8 16k;           //缓存php生成的页面内容，8个16k
    fastcgi_buffer_size 32k;        //缓存php生产的头部信息
    fastcgi_connect_timeout 300;    //连接PHP的超时时间
    fastcgi_send_timeout 300;       //发送请求的超时时间
    fastcgi_read_timeout 300;       //读取请求的超时时间
    location ~ \.php$ {
        root    html;
        fastcgi_pass 127.0.0.1:9000;
        fastcgi_index index.php;
        include fastcgi.conf;
    }
```



## 安装服务端LNMP环境（续2）

知识讲解

- 启动服务

```
[root@zabbix server ~]# systemctl start mariadb
```

```
[root@zabbix server ~]# systemctl start php-fpm
```

```
[root@zabbix server ~]# /usr/local/nginx/sbin/nginx
```

- 测试页面

```
[root@zabbix server ~]# cat /usr/local/nginx/html/test.php
```

```
<?php
```

```
$i=33;
```

```
echo $i
```

```
?>
```



## 部署Zabbix

# 编译安装Zabbix服务

知识讲解

- 源码安装软件

```
[root@zabbix server ~]# yum -y install net-snmp-devel \
> curl-devel libevent-devel
[root@zabbix server ~]# tar -xf zabbix-3.4.4.tar.gz
[root@zabbix server ~]# cd zabbix-3.4.4/
[root@zabbix server zabbix-3.4.4]# ./configure --enable-server \
> --enable-proxy --enable-agent --with-mysql=/usr/bin/mysql_config \
> --with-net-snmp --with-libcurl
[root@zabbix server zabbix-3.4.4]# make && make install
```



# 初始化Zabbix

知识讲解

- 创建数据库与数据库账户

```
[root@zabbix server ~]# mysql
mysql> create database zabbix character set utf8;
mysql> grant all on zabbix.* to zabbix@'localhost' identified by 'zabbix';
[root@zabbix server ~]# cd /root/zabbix-3.4.4/database/mysql/
[root@zabbix server mysql]# mysql -uzabbix -pzabbix zabbix < schema.sql
[root@zabbix server mysql]# mysql -uzabbix -pzabbix zabbix < images.sql
[root@zabbix server mysql]# mysql -uzabbix -pzabbix zabbix < data.sql
```

- 上线Zabbix页面

```
[root@zabbix server ~]# cd /root/lnmp_soft/zabbix-3.4.4/frontends/php/
[root@zabbix server php]# cp -a * /usr/local/nginx/html/
[root@zabbix server php]# chmod -R 777 /usr/local/nginx/html/*
```



## 初始化Zabbix ( 续1 )

- 修改配置文件，启动zabbix server服务

知识讲解

```
[root@zabbix server ~]# useradd zabbix //不创建用户无法启动服务
[root@zabbix server ~]# vim /usr/local/etc/zabbix_server.conf
DBHost=localhost //数据库主机
DBName=zabbix //设置数据库名称
DBUser=zabbix //设置数据库账户
DBPassword=zabbix //设置数据库密码
LogFile=/var/log/zabbix/zabbix_server.log //设置日志
[root@zabbix server ~]# zabbix_server //启动服务
```



## 初始化Zabbix ( 续2 )

- 修改配置文件，启动zabbix agent ( 被监控时使用 )

知识讲解

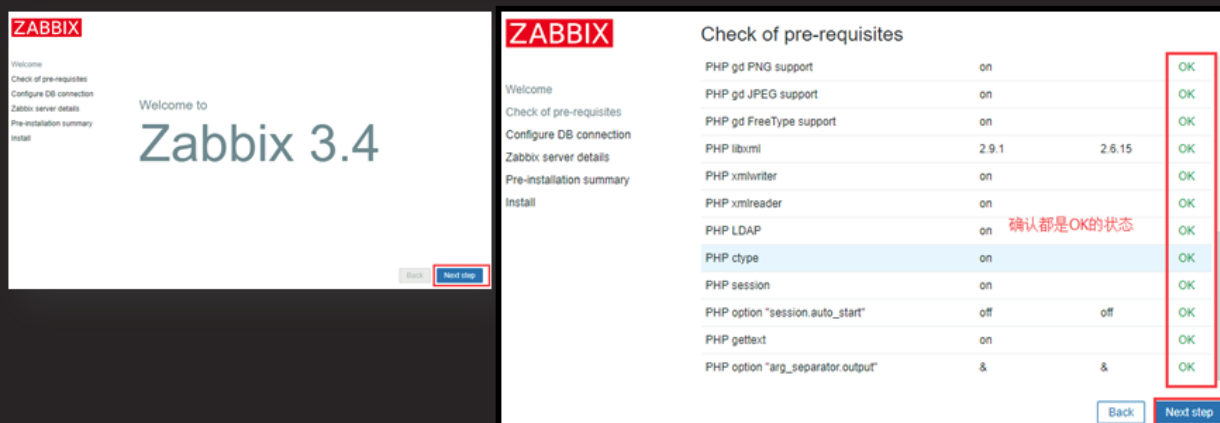
```
[root@zabbix server ~]# vim /usr/local/etc/zabbix_agentd.conf
Server=127.0.0.1,192.168.2.5 //设置监控服务器IP
ServerActive=127.0.0.1,192.168.2.5 //主动监控服务器IP
Hostname=zabbix_server //设置本机主机名
LogFile=/tmp/zabbix_server.log //设置日志文件
UnsafeUserParameters=1 //是否允许自定义key
[root@zabbix server ~]# zabbix_agentd //启动监控agent
```



## 初始化Zabbix ( 续3 )

- 初始化Web管理页面 ( 浏览器访问web )
- Zabbix初始化时会检查环境是否满足要求

知识讲解



## 初始化Zabbix ( 续4 )

- 根据检查的报警提示, 修改系统环境

知识讲解

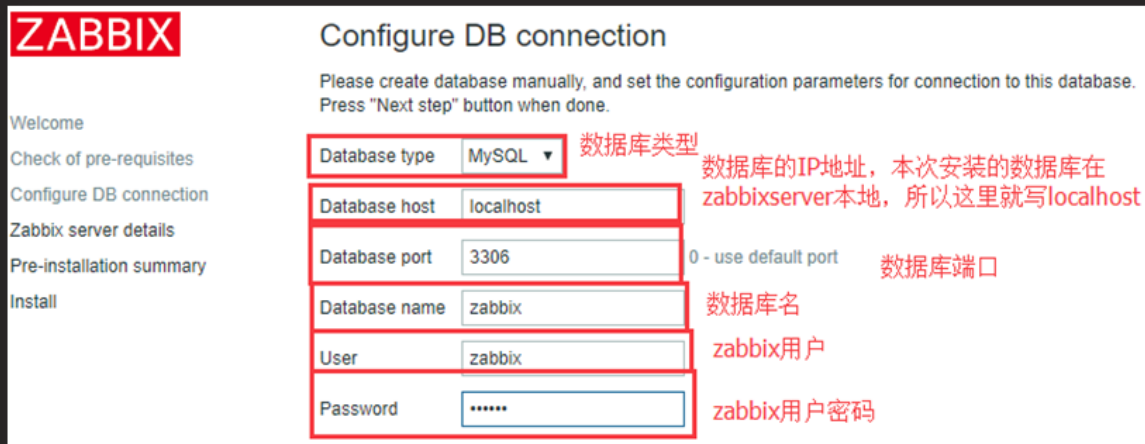
```
[root@zabbix server ~]# yum -y install php-gd php-xml
[root@zabbix server ~]# yum localinstall php-bcmath-5.4.16-42.el7.x86_64.rpm
[root@zabbix server ~]# yum localinstall php-mbstring-5.4.16-42.el7.x86_64.rpm
[root@zabbix server ~]# vim /etc/php.ini
date.timezone = Asia/Shanghai           //设置时区
max_execution_time = 300                 //最大执行时间, 秒
post_max_size = 32M                     //POST数据最大容量
max_input_time = 300                    //服务器接收数据的时间限制
memory_limit = 128M
[root@zabbix server ~]# systemctl restart php-fpm
```



## 初始化Zabbix (续5)

- 根据提示修改数据库信息

知识讲解



**ZABBIX** Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

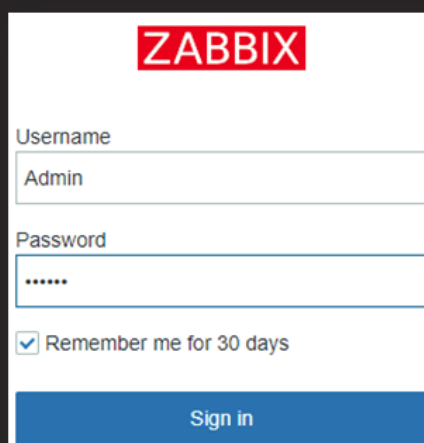
Database type	MySQL	数据库类型
Database host	localhost	数据库的IP地址, 本次安装的数据库在zabbixserver本地, 所以这里就写localhost
Database port	3306	0 - use default port 数据库端口
Database name	zabbix	数据库名
User	zabbix	zabbix用户
Password	*****	zabbix用户密码



## 初始化Zabbix (续5)

- 默认登陆账户admin, 默认密码zabbix
- 设置中文环境 (推荐英文, 中文小部分为乱码)

知识讲解



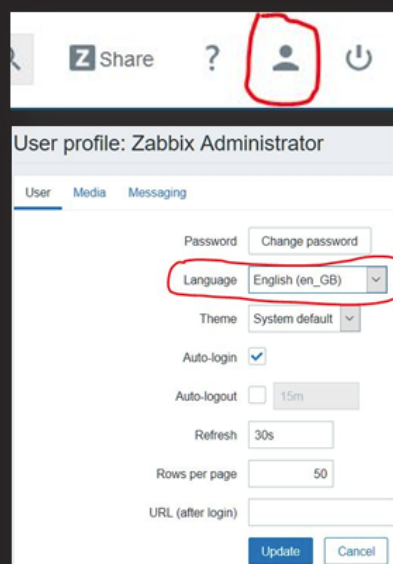
**ZABBIX**

Username  
Admin

Password  
\*\*\*\*\*

☒ Remember me for 30 days

Sign in



User profile: Zabbix Administrator

User Media Messaging

Password  Change password

Language

Theme

Auto-login ☒

Auto-logout ☐ 15m

Refresh

Rows per page

URL (after login)

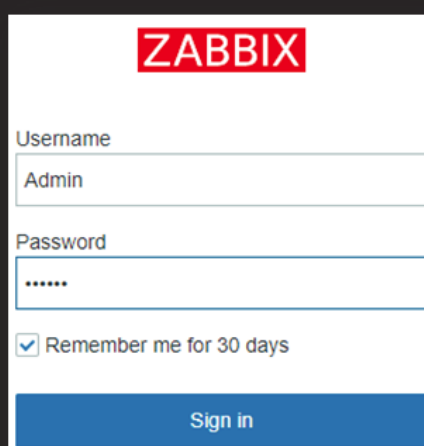
Update Cancel



## 初始化Zabbix ( 续5 )

- 默认登陆账户admin，默认密码zabbix
- 设置中文环境（推荐英文，中文小部分为乱码）

知识讲解



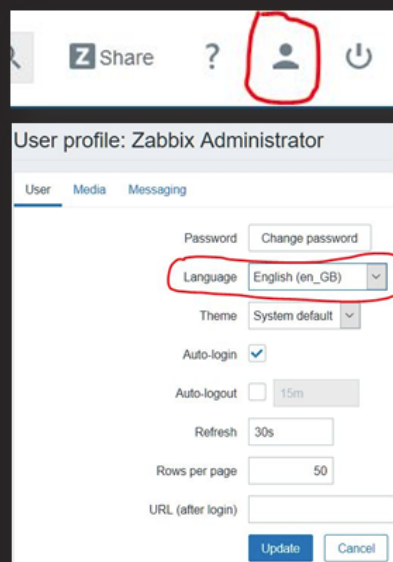
**ZABBIX**

Username  
Admin

Password  
\*\*\*\*\*

☒ Remember me for 30 days

Sign in



User profile: Zabbix Administrator

User Media Messaging

Password  Change password

Language **English (en\_GB)**

Theme System default

Auto-login ☒

Auto-logout ☐ 15m

Refresh 30s

Rows per page 50

URL (after login)

Update Cancel



## 被监控主机安装Zabbix

- 在2.100和2.200做相同操作（以web1为例）

```
[root@web1 ~]# useradd -s /sbin/nologin zabbix
[root@web1 ~]# yum -y install gcc pcre-devel
[root@web1 ~]# tar -xf zabbix-3.4.4.tar.gz
[root@web1 ~]# cd zabbix-3.4.4/
[root@web1 zabbix-3.4.4]# ./configure --enable-agent
[root@web1 zabbix-3.4.4]# make && make install
```

- 拷贝启动脚本（可选操作）

```
[root@web1 zabbix-3.4.4]# cd misc/init.d/fedora/core
[root@web1 zabbix-3.4.4]# cp zabbix_agentd /etc/init.d/
```

知识讲解





## 被监控主机安装Zabbix (续2)

知识讲解

- 修改Agent配置文件并启动服务

```
[root@web1 ~]# vim /usr/local/etc/zabbix_agentd.conf
Server=127.0.0.1,192.168.2.5           //谁可以监控本机 ( 被动监控 )
ServerActive=127.0.0.1,192.168.2.5    //谁可以监控本机 ( 主动监控 )
Hostname=zabbix_client_web1           //被监控端自己的主机名
EnableRemoteCommands=1
//监控异常后，是否允许服务器远程过来执行命令，如重启某个服务
UnsafeUserParameters=1                //是否允许自定义key监控
[root@web1 ~]# zabbix_agentd           //启动agent服务
```



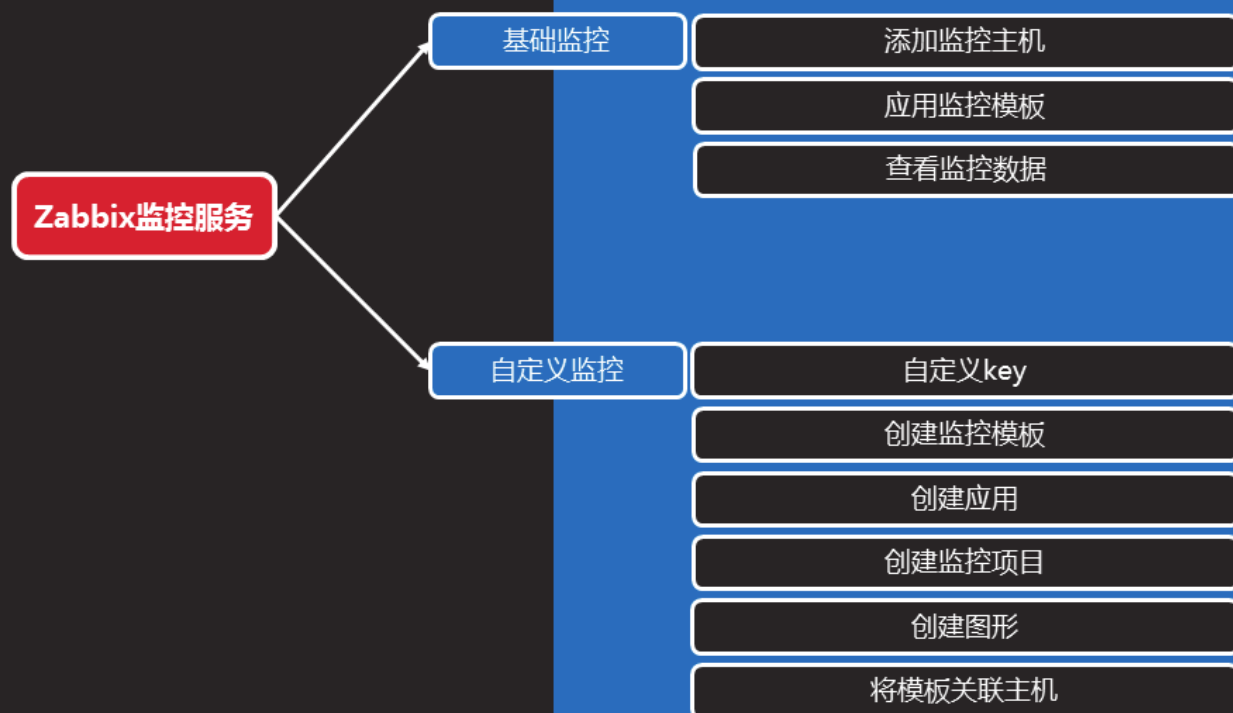
## 案例2：部署Zabbix监控平台

课堂练习

- 安装LNMP环境
- 源码安装Zabbix
  - 安装监控端主机，修改基本配置
- 初始化Zabbix监控Web页面
  - 修改PHP配置文件，满足Zabbix需求
- 安装被监控端主机，修改基本配置



# Zabbix监控服务

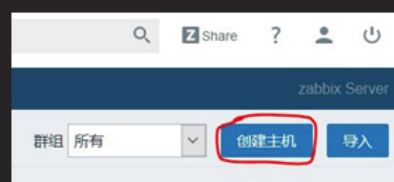


## 基础监控

## 添加监控主机

知识讲解

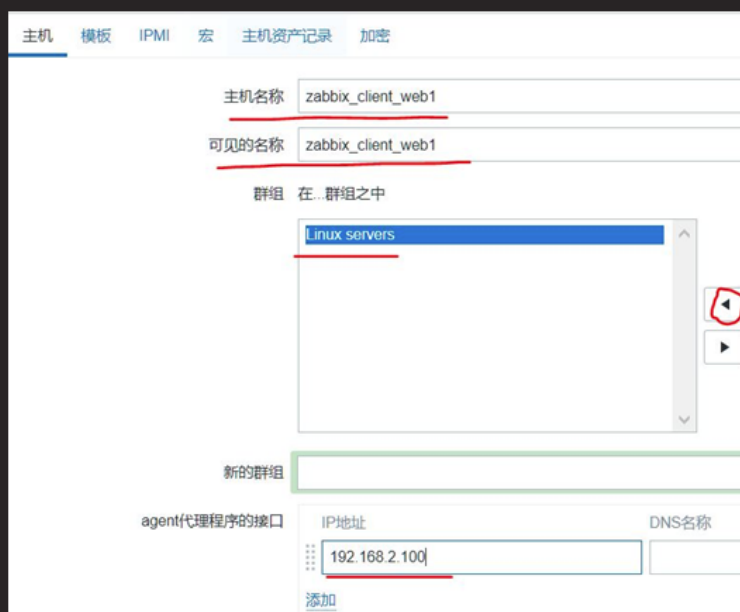
- Host ( 主机 ) 是监控的基本载体
- Zabbix所有监控都是基于Host
- 通过Configuration→Hosts→Create Host创建
  - 注意：设置中文环境后，中英文差异



## 添加监控主机（续1）

知识讲解

- 根据提示输入
  - Host name
  - Visible name
  - Groups in groups
  - IP address
  - 其他默认即可



## 应用监控模板

- 为主机添加关联的监控模板
  - 在“Templates” 模板选项卡页面中
  - 找到Link new templates , select选择合适的模板添加
  - 这里我们选择Template OS Linux模板

知识讲解



## 查看监控数据

- 可以点击"Monitoring" -> "Latest data"
- 在过滤器中填写条件，根据群组 and 主机搜索即可

知识讲解

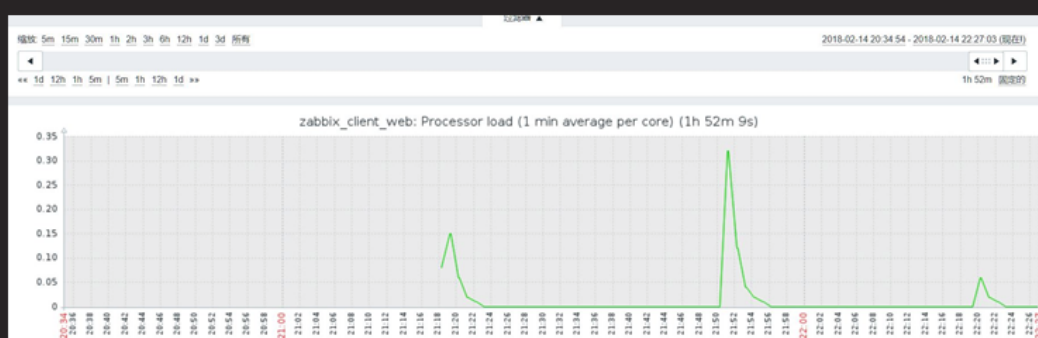


## 查看监控数据（续1）

- 找到需要的数据，点击后面的Graph

知识讲解

<input type="checkbox"/> Processor load (1 min average per core)	2018-02-14 22:27:27	图形
<input type="checkbox"/> Processor load (5 min average per core)	2018-02-14 22:27:28	图形
<input type="checkbox"/> Processor load (15 min average per core)	2018-02-14 22:27:26	图形



## 案例3：配置及使用Zabbix监控系统

- 使用Zabbix监控平台监控Linux系统
  - 监控CPU
  - 监控内存
  - 监控进程
  - 监控网络流量
  - 监控硬盘

课堂练习

# 自定义监控

## 自定义key

- 被监控端修改Agent配置文件

```
[root@web1 ~]# vim /usr/local/etc/zabbix_agentd.conf  
Include=/usr/local/etc/zabbix_agentd.conf.d/ //加载配置文件目录
```

- 创建自定义key

```
[root@web1 ~]# cd /usr/local/etc/zabbix_agentd.conf.d/  
[root@web1 zabbix_agentd.conf.d]# vim count.line.passwd  
UserParameter=count.line.passwd,wc -l /etc/passwd | awk '{print $1}'  
//自定义key语法格式  
//UserParameter=自定义key名称,命令
```

## 自定义key ( 续1 )

知识讲解

- 重启Agent

```
[root@web1 ~]# killall zabbix_agentd
```

```
[root@web1 ~]# zabbix_agentd
```

- 测试自定义key是否生效

```
[root@web1 ~]# zabbix_get -s 127.0.0.1 -k count.line.passwd  
21
```

//如提示Check access restrictions in Zabbix agent configuration  
//则需要检查配置文件：

```
[root@web1 ~]# vim /usr/local/etc/zabbix_agentd.conf
```

```
Server=127.0.0.1,192.168.2.5
```

```
ServerActive=127.0.0.1,192.168.2.5
```



## 创建监控模板

- 登录监控服务器Web管理页面
  - 选择Configuration→Templates创建模板

知识讲解



# 创建监控模板（续1）

- 设置模板名称与组名称
  - Template name
  - Visible name
  - New group

知识讲解

The screenshot shows the ZABBIX web interface for creating a template. The 'Template' tab is selected in the top navigation bar. The 'Template name' and 'Visible name' fields are both set to 'count.line.passwd'. The 'New group' field is set to 'count-passwd'. The 'Groups' section is empty.



## 创建监控项目

- 与创建应用一样，创建项目
  - Configuration→Templates→Items→Create item

知识讲解

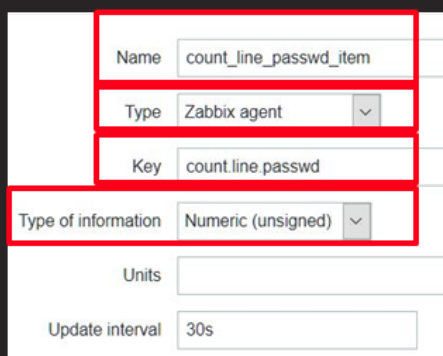
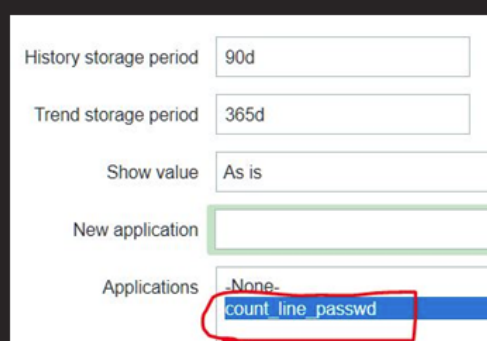
<input type="checkbox"/> Name ▲	Applications	Items	Triggers	Graphs
<input type="checkbox"/> count.line.passwd	Applications 1	Items 1	Triggers	Graphs
<input type="checkbox"/> Template App Apache Tomcat JMX	Applications 5	Items 32	Triggers 5	Graphs 4

创建监控项



## 创建监控项目（续1）

- 设置项目参数
  - 项目名称
  - 自定义key（必须与配置文件一致）
  - 应用集选择刚刚创建的应用(Application)

知识讲解



## 创建图形

- 与监控项目类似，为监控数据创建图形

<input type="checkbox"/> 名称 ▲	应用集	监控项	触发器	图形
<input type="checkbox"/> count.line.passwd	应用集 1	监控项 1	触发器 5	<input checked="" type="checkbox"/> 图形 4
<input type="checkbox"/> Template App Apache Tomcat JMX	应用集 5	监控项 32	触发器 5	<input type="checkbox"/> 图形 4

创建图形

知识讲解



## 创建图形（续1）

- 设置图形参数
  - 填写名称
  - 图形类别（以此为线条、填充图、饼图、分割饼图）
  - 添加监控项目

知识讲解

名称

宽

高

图形类别 正常 ▾

查看图例 ☒

查看工作时间 ☒

查看触发器 ☒

监控项 名称

1: count.line.passwd: count\_line\_passwd\_item

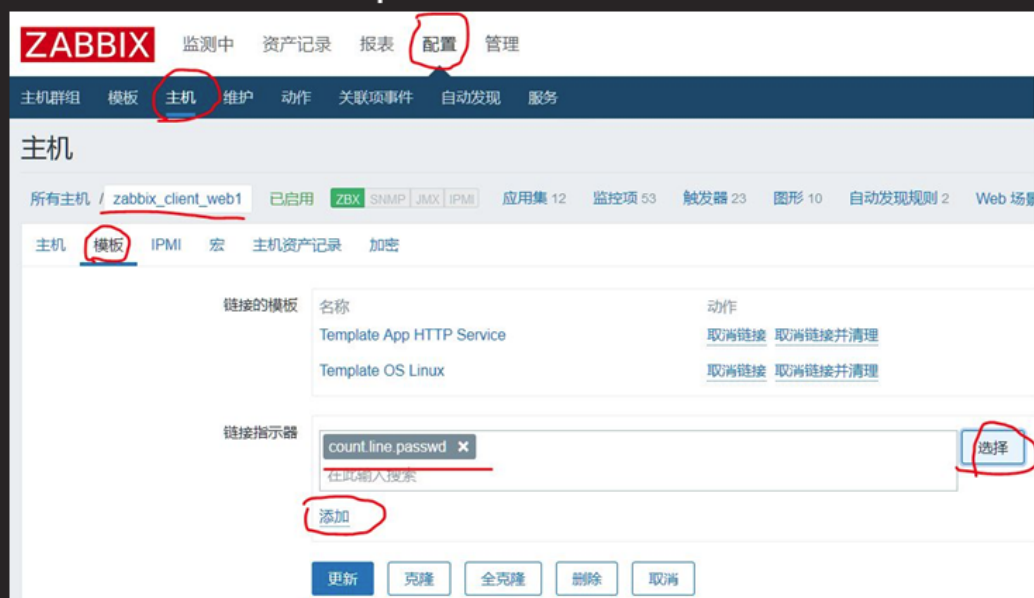
添加

添加 取消

## 将模板关联主机（续1）

- 点击Templates，select选项监控项目，add添加
- 添加完成后，点击Update更新主机配置

知识讲解



## 案例4：自定义Zabbix监控项目

课堂练习

- 使用Zabbix监控Linux服务器的账户数量
  - 使用内置模板监控Linux
  - 创建自定义key
  - 创建监控项目
  - 创建监控图形
  - 将自定义监控模板关联到主机，实现监控目标



### 总结和答疑

总结和答疑

自定义监控错误

问题现象

故障分析及排除

