

系统监控-安全补丁

Smartbi添加了一种安全补丁机制，能够在不更新war包的前提下，只更新安全补丁。

该安全补丁可以修复一些系统漏洞，避免发生由于漏洞引起的安全事故。

在Smartbi中安装安全补丁涉及以下两个文件：

补丁工具包：补丁工具包是用来上传安全补丁文件的载体，通过此工具包可在Smartbi服务器上上传安全补丁文件。（当前补丁工具包最新版本为1.0）

- 更新流程
 - 判断系统版本
 - 安装补丁工具包
 - 更新补丁工具包
 - 更新安全补丁文件
 - 在线更新
 - 手动更新
- 说明事项
 - 授权IP地址访问config页面
 - 限制config页面文件访问路径

安全补丁文件：安全补丁文件中包含了目前我们检测修复的安全漏洞程序。Smartbi会不定期在官网更新安全补丁文件，用户可下载此文件，并直接更新到Smartbi服务器中，已解决已知的一些安全漏洞。

根据实际情况，在官网下载补丁工具包或安全补丁文件，官网如图：

SMARTBI 思迈特软件

首页产品功能解决方案客户及案例体验中心服务关于

会员中心

Smartbi安全补丁包下载

Smartbi 安全补丁包

安全补丁文件

产品安全补丁更新文件，跟工具包配套使用，会不定期更新

更新日期：2020-04-20

大小：10KB

补丁使用说明

选择您当前软件版本

V8系列

V8.5.655以前

V8.5.655以后

V9系列

V9.3.000以前

V9.3.000以后

其他

其他版本

立即下载

补丁工具包 V1

补丁更新记录

补丁工具包更新记录

咨询

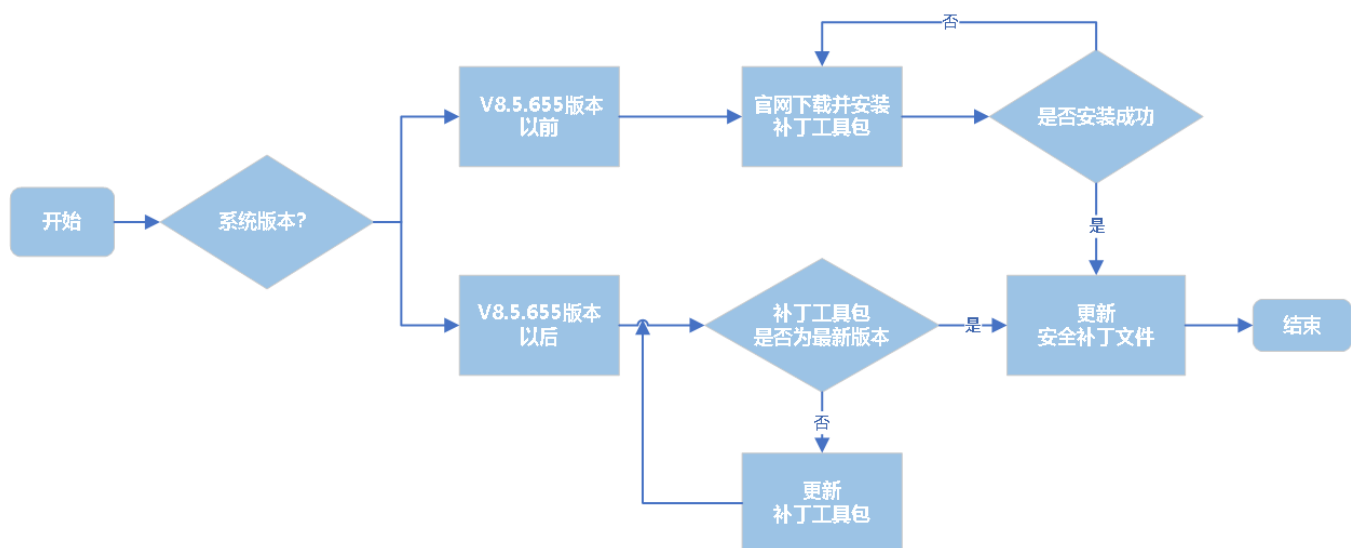
支持

电话

邮箱

更新流程

更新安全补丁的具体流程如下：



按照上述流程图进行操作即可。

判断系统版本

在系统的右上角上点击 **用户名称 > 关于**，判断系统版本是否为V8. 5. 655，再进行下一步操作：

- 如果当前系统版本为V8. 5. 655以前的版本，则表明系统没有默认带有补丁工具包，需要先到官网安装补丁工具包，具体请参考 [安装补丁工具包](#)。
- 如果当前系统版本为V8. 5. 655及以后的版本，则表明系统默认带有补丁工具包，需要检查补丁工具包版本是否需要更新，具体请参考 [更新补丁工具包](#)。



安装补丁工具包

前提：当前系统版本为V8. 5. 655以前的版本，且未安装过安全补丁。

具体操作如下：

- 1、去官网根据版本下载最新的补丁工具包。下载地址：[Smartbi安全补丁包下载](#)。
- 2、参考文档 [扩展包部署](#)，将获取到的安装补丁工具包部署到Smartbi应用服务器上。

3、完成部署后，重启Smartbi应用服务器，验证是否安装成功。可在系统监控面板查看是否补丁工具包安装成功：

服务器： 设置

当前服务器

导出所有

概述

监视

线程

性能

网络

日志

会话

缓存

堆打印

对象池

类查找

扩展包

所有字体

wsdl地址

log4j信息

SQL监控

安全补丁

当前版本

安全工具包： 1.0

安全补丁： 没有找到 [在线更新](#) [手动更新](#)

已应用安全补丁

使用说明

1、产品最新版本已默认包含相关安全信息的修复

2、未更新最新版本的用户可以去官网下载补丁工具包和补丁文件

3、官网会不定期更新工具包和补丁文件

4、更新的补丁文件需要跟对应版本的工具包匹配使用，下载时请注意是否需要更新工具包

5、补丁文件更新成功后立即生效，无需重启服务器

安装完补丁工具包后，补丁工具包版本已是最新的，接下来 [更新安全补丁文件](#) 即可。

更新补丁工具包

前提：当前系统版本为V8.5.655及以后的版本，或曾经安装过补丁工具包但不确定补丁工具包是否为最新。

具体操作如下：

1、登录服务器，打开系统监控面板。查看当前补丁工具包版本，再与官网上最新的补丁工具包对比，判断当前补丁工具包的版本是否为最新。

- 如果是最新，则只需更新安全补丁文件。详情请参考 [更新安全补丁文件](#)。
- 如果不是最新，则需先更新补丁工具包，之后再更新安全补丁文件。具体请看下个步骤。

服务器： 设置

当前服务器

导出所有

概述

监视

线程

性能

网络

日志

会话

缓存

堆打印

对象池

类查找

扩展包

所有字体

wsdl地址

log4j信息

安全补丁

当前版本

安全工具包： 1.0 **当前版本工具包版本需要为最新**

安全补丁： 2020-04-14 17:04:00 [在线更新](#) [手动更新](#)

已应用安全补丁

- 限制配置界面及管理页面的访问地址 (Patch.20200414 @2020-04-14)
- 未授权登录配置界面、SQL注入 (Patch.20200325 @2020-03-25)
- XSS 攻击 (Patch.20191218 @2019-12-18)
- SQL注入、访问其它文件 (PATCH_20191212 @2019-12-12)

使用说明

1、产品最新版本已默认包含相关安全信息的修复

2、未更新最新版本的用户可以去官网下载补丁工具包和补丁文件

3、官网会不定期更新工具包和补丁文件

4、更新的补丁文件需要跟对应版本的工具包匹配使用，下载时请注意是否需要更新工具包

5、补丁文件更新成功后立即生效，无需重启服务器

2、去官网根据版本下载最新的补丁工具包。下载地址：[Smartbi安全补丁包下载](#)。

3、参考文档 [扩展包部署](#)，将获取到的安装补丁工具包部署到Smartbi应用服务器上。

4、完成部署后，重启Smartbi应用服务器，再次打开系统监控界面，查看补丁工具包的版本信息，验证是否更新成功。



更新安全补丁文件

前提：安全补丁文件需要更新，且当前补丁工具包的版本为最新。

更新安全补丁文件的方法有两种，分别为“在线更新”和“手动更新”。



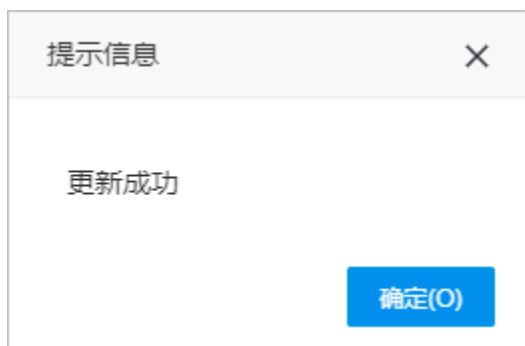
补丁工具包的日期为2020年3月10号之后的才支持“在线更新”功能，并且“在线更新”功能要求当前系统环境能正常访问外网，否则会提示更新失败。

在线更新

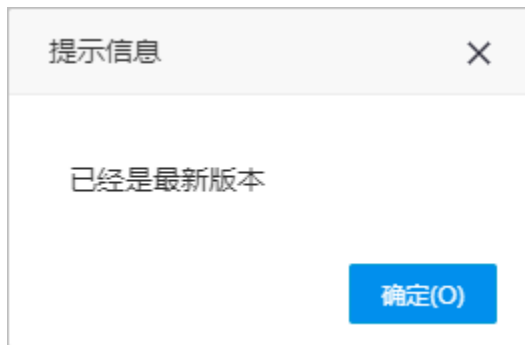
打开“系统监控 > 安全补丁”界面，点击 **在线更新** 按钮，系统会到官网上自动获取最新的安全补丁文件进行更新。



更新完成提示如下：

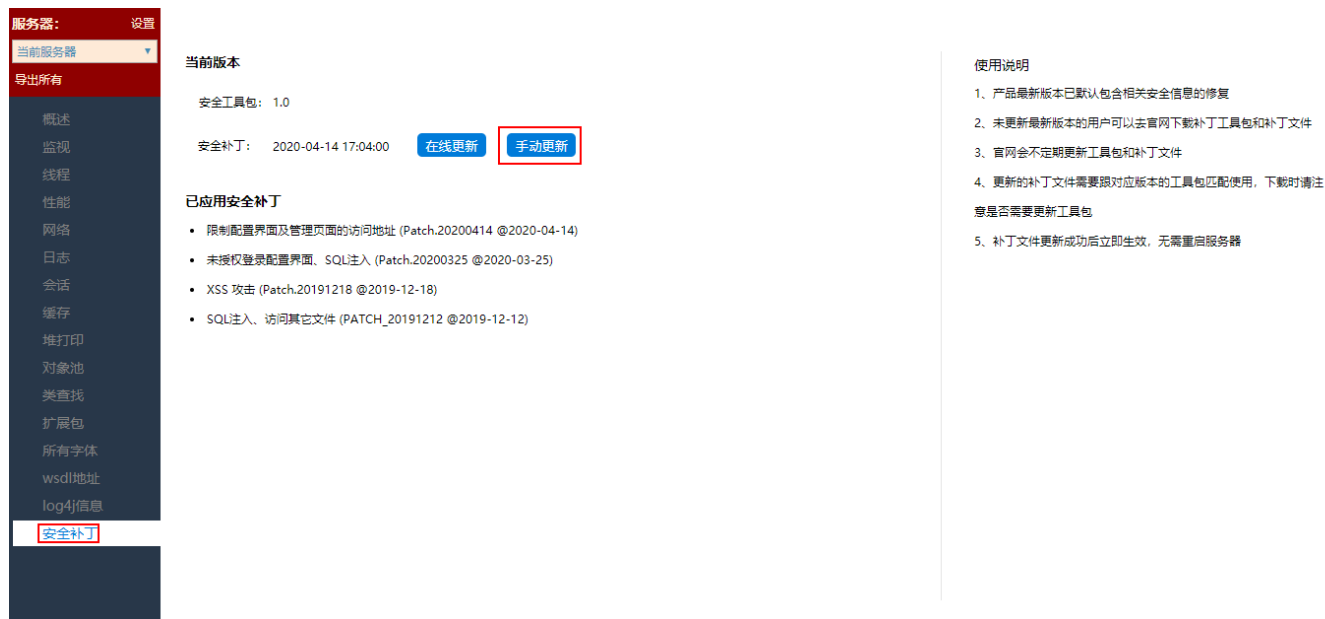


如果当前版本的补丁更新文件已为最新版本，则会提示“已经是最新版本”。

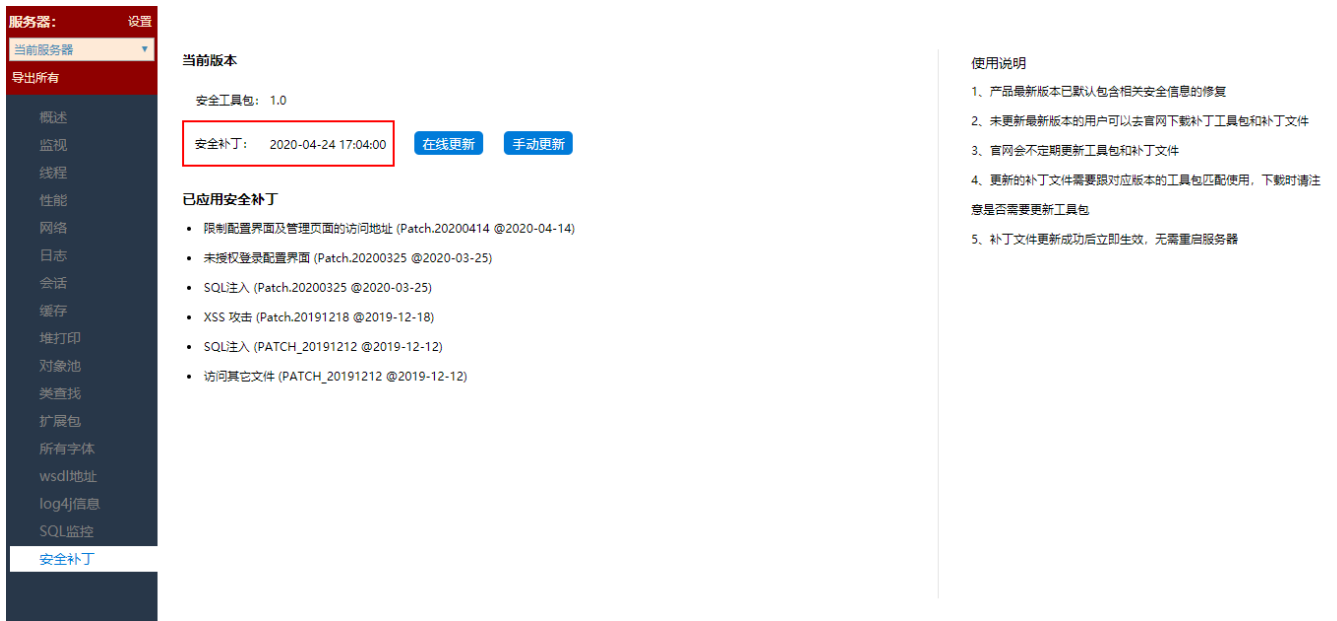


手动更新

- 1、先从官网下载最新的补丁更新文件。下载地址：[Smartbi安全补丁包下载](#)。
- 2、打开“系统监控 > 安全补丁”界面，点击 **手动更新** 按钮，选择刚下载的安全补丁文件并上传。



- 3、安全补丁文件更新成功后立即生效，不需要重启服务器。重新进入“安全补丁”界面查看其版本信息，验证是否更新成功。



说明事项

授权IP地址访问config页面

如果访问smartbi的config和monitor配置页面时，提示“未授权的IP地址，该页面需要授权IP地址才能访问，请联系管理员添加”。

需要找到Smartbi安装目录下的smartbi.properties文件配置你需要允许的IP。

 可以使用英文逗号分隔设置多个IP属性，例如smartbi.allowedConfigIps=10.10.23.11,10.10.201.1-10.10.201.254,10.10.202.0/24

配置说明如下：

类别	方法	示例
允许所有IP	*号代表全部IP。 注：这里不支持10.10.202.*	smartbi.allowedConfigIps=*
精确指定某个ip	直接设置对应的ip地址即可。	smartbi.allowedConfigIps=10.10.101.11
指定多个ip地址	各个ip地址以英文逗号分隔。	smartbi.allowedConfigIps=10.10.101.11,10.10.101.21,10.10.101.31
指定某个ip段地址	指定ip段地址区间，中间用减号（-）连接。	smartbi.allowedConfigIps=10.10.101.0-10.10.101.255
指定ip支持标准的掩码	使用 标准CIDR 格式。计算方法可参考： 计算方法	smartbi.allowedConfigIps=10.10.23.0/24

注意：

- 1、修改后只需刷新页面即可，无需重启服务器。
- 2、allowedConfigIps这个属性控制的是config界面；如果控制的是monitor界面，对应的属性是allowedMonitorIps。
- 3、“smartbi.”这个前缀是上下方路径。如果项目不是使用/smartbi/vision访问，而是/smartbiNew/vision/index.jsp这样，则前缀应当是“smartbiNew.”，即smartbiNew.allowedConfigIps。
- 4、如果服务器为Linux，可以将configip.sh上传到smartbi.properties所在目录，并通过 `chmod 755 configip.sh` 修改为可执行，再运行 `./configip.sh` 通过命令行修改。

命令行修改工具：[configip.sh](#)

限制config页面文件访问路径

新建任务，任务类型选择定制，粘贴如下脚本代码：

任务基本信息

任务名称: * 修改访问路径

任务别名: 修改访问路径

任务描述:

任务类型: 定制

自定义设置

1

查看运行脚本(V) 测试运行(T) 保存(S) 关闭(C)

```

//
importPackage(Packages.smartbi.sdk.service.datasource);

/**
 * linux/data/tomcat/binwindowsD:\\\\tomcat\\\\bin
 */
var path="D:\\\\tomcat\\\\bin";

////////////////////////key////////////////////////////////////
//t_systemconfigkey
var key = "JSP_CHOOSER_ROOT_PATH";

//key
var querySql="select c_value from t_systemconfig where c_key='"+key+"'";

//keyvalue
var insertSql="insert into t_systemconfig(c_key, c_value) values('"+key+"','"+path+"')";

//value
var updateSql="update t_systemconfig set c_value='"+path+"' where c_key='"+key+"'";

//ID
var dsId="DS.SYSTEM";

var datasrcService = new DataSourceService(connector);

try {
    var res = datasrcService.executeNoCacheable(dsId, querySql);

    //logger.info(""+res.getRowsCount());
    //key
    if(res.getRowsCount()>0){
        datasrcService.executeUpdate(dsId, updateSql);
    }else{
        //key
        datasrcService.executeUpdate(dsId, insertSql);
    }
    //
    connector.remoteInvoke("CompositeService", "clearCache", []);
} catch (e) {
    logger.error(e);
}

```

修改第9行的路径设置，保存后运行即可：

任务配置【修改访问路径】

任务名称: *

修改访问路径

任务别名:

修改访问路径

任务描述:

任务类型:

定制

自定义设置

```
1 //定制任务脚本
2 importPackage(Packages.smartbi.sdk.service.datasource);
3
4 /**
5  * 配置路径, 请按实际情况修改
6  * linux格式: /data/tomcat/bin; windows格式需要转义处理: D:\\\\tomcat\\\\bin
7  */
8 var path="D:\\\\tomcat\\\\bin";
9
10
11 //以下部分为判断是否存在key, 不存在则新增, 存在则更新值, 默认即可不需要修改
12 //t_systemconfig表中的key
13 var key = "JSP_CHOOSER_ROOT_PATH";
14
15 //查询知识库中是否存在key的语句
16 var querySql="select c_value from t_systemconfig where c_key='"+key+"'";
17
18 //插入key和value语句
19 var insertSql="insert into t_systemconfig(c_key, c_value) values('"+key+"','"+path+"')";
20
```

查看运行脚本(V)

测试运行(T)

保存(S)

关闭(C)

说明: 该逻辑会先判断对应的key是否存在, 不存在则新增, 存在则更新value值;

如果不需要清空缓存, 可手动屏蔽第41行代码, 一般不清空缓存不会立刻生效

任务配置【修改访问路径】

任务名称: *

修改访问路径

任务别名:

修改访问路径

任务描述:

任务类型:

定制

自定义设置

```
25 var dsId="DS.SYSTEM知识库";
26
27 var datasrcService = new DataSourceService(connector);
28
29 try {
30     var res = datasrcService.executeNoCacheable(dsId, querySql);
31
32     //logger.info("返回行数: "+res.getRowsCount());
33     //如果已经存在key, 则更新
34     if(res.getRowsCount()>0){
35         datasrcService.executeUpdate(dsId, updateSql);
36     }else{
37         //如果不存在key, 则新增
38         datasrcService.executeUpdate(dsId, insertSql);
39     }
40     //清空缓存, 如果不需要清空缓存, 可手动屏蔽, 一般不清空缓存不会立刻生效
41     connector.remoteInvoke("CompositeService", "clearCache", []);
42 } catch (e) {
43     logger.error(e);
44 }
```

查看运行脚本(V)

测试运行(T)

保存(S)

关闭(C)