## Aegisshield前端详细设计

（1）整体框架流程：

登录模块->控制操作模块

登录模块：

用户进行登录

控制操作模块：

用户管理模块：查看、删除、修改用户密码，添加用户

权限：管理账户

规则管理模块：查看、删除、添加规则

权限：管理账户和普通用户

日志管理模块：

1. 总体日志信息：sensor数，Alert数等
2. 入侵图形分析：如，时间-Alerts数等
3. 应用层协议分析：迅雷的协议，QQ等

权限：管理账户和普通用户

（2）数据库

1用户ag\_users

$DB->Execute("CREATE TABLE ag\_users ( id INT(10) NOT NULL AUTO\_INCREMENT, username VARCHAR(20) NOT NULL, passwd VARCHAR(100) NOT NULL, privilege INT(1) NULL, name VARCHAR(60) NULL, phone VARCHAR(30) NULL, email VARCHAR(100) NOT NULL, created DATETIME NULL, PRIMARY KEY(id))");

用户id（key），登录名称，登录密码，显示名称，权限，电话，邮箱，创建日期

2规则ag\_rules

id | type | rule | proto

type主要是为了区别应用层协议分析和Snort

proto是应用层协议分析专用的，说明这个rule是什么应用层软件

3应用层协议数据

4 snort数据表

（3）具体类与页面设计

1具体函数类

主要有以下几个类，ag\_session(会话类)，page（页面管理类），user（用户管理类），alerts（警告类），utility（工具类），graph\_data（图形数据类）

2页面文件说明：

index.php

:默认主页面，登录界面。

登录成功将根据不同用户跳转到，admin.php页面（管理员账户页面）或者user.php页面（一般用户界面）

登录失败，如用户名密码错误将在此页面上显示登录错误。

admin.php

管理员主操作页面。

管理员账户登录成功后，会跳转到该页面。

user.php

普通用户操作页面。

普通用户登录成功后，会跳转到该页面

includes目录下：

config.php

配置文件，初始化后面需要的类，及进行数据库的链接，保存会话

connection\_setting.php

数据库连接配置文件，设置数据库账户，密码，数据库名称等

session\_settings.php

会话变量的配置，会话时间及会话回收时间

classes目录下

session\_class.php

class ag\_session（会话类）

作用：管理系统会话，控制用户登录，注销已经数据库访问的链接

变量：

$ag\_session\_id 会话id

$session\_time 会话时间

$session\_gc\_time 会话垃圾回收时间

$DB 数据访问的链接

重要函数：

ag\_session($session\_time,$session\_gc\_time,$DB)

:构造函数

auth\_get\_status()

:如果可能（如存在cookie）自动登录

login($username,$password)

:登录

set\_log($user\_id,$type\_auth)

:记录登录日志

logout()

:注销

page\_class.php

class page (页面类）

作用：html语言页面的显示，方便管理页面头部和底部

函数：

page()

:构造函数

getHeader($interface,$title)

:返回html头部代码，$title为title名称，$interface为 导入的style文件

getFooter()

:返回</body></html>

get\_credits()

:返回版权信息

utility\_class.php

class utility (工具类）

作用：提过一些常用的功能

函数：

utility()

:构造函数

get\_logo($link)

:返回logo图片的链接

user\_class.php

class user (用户类）

作用：管理用户,提供添加用户，修改用户信息，删除用户操作

变量：

var $DB;

重要函数：

user($DB)

:构造函数

exist\_user($id)

:判定某个用户是否存在

check\_user($user)

:判定用户名是否有效

check\_password($password)

:判定密码是否有效

check\_email($email)

：判断邮箱名是否有效

id\_to\_user($id)

:根据用户id返回用户名

user\_to\_id($user)

:根据用户名返回用户id

add\_user($user\_array)

:添加用户

change\_password($id,$password)

:修改用户密码

del\_user($id)

:删除用户

set\_user($user\_array)

:修改用户的信息

其他(待更新）

rule\_class.php

class rule (规则类)

作用:管理规则，提供规则的添加，修改及删除

变量：

$DB

函数：

rule($DB)

:构造函数

exist\_rule($id)

:判断该规则是否存在

add\_rule($rule\_array)

:添加规则

change\_rule($id,$rule)

:修改规则

del\_rule($id)

:删除规则

get\_rule($id)

:返回该条规则

get\_rules()

:返回显示数据库存在的规则

option\_list\_admin($id)

:admin页面中的规则操作列表

option\_list\_user($id)

:user页面的规则操作列表

/\* 该部分应该会去掉

language\_class.php

class language(语言类，扩展可以使用英语和中文两种)

作用：管理显示语言

函数：

language()

:构造函数

getLanguages()

:返回存在的语言

languagesList($lang)

:返回语言的列表

\*/

alerts\_class.php

class alerts(警报类）

作用：提供显示多种警报形式，如总警报数，某个ip的警报数，某个端口的警报数

变量：

$DB;

函数：

alerts($DB)

:构造函数

sensorCnt()

:传感器数

eventCnt()

:总警报数

eventBySensor($sensorID)

:某个传感器上的警报数

uniqueAlertCnt()

:总的警报类型数

uniqueAlertCntBySensor($sensorID)

:某个传感器上的总的警报类型数

uniqueIpDstCnt()

:产生警报的目的ip地址数目

uniqueIpSrcCnt()

:产生警报的源ip地址数目

uniqueLinkCnt()

:产生警报的链接数

uniqueSrcPortCnt()

:产生警报的源端口数目

uniqueDstPortCnt()

:产生警报的目的端口数目

uniqueTcpSrcPortCnt()

:产生警报的Tcp源端口数目

uniqueTcpDstPortCnt()

:产生警报的Tcp目的端口数

uniqueUdpSrcPortCnt()

:产生警报的Udp源端口数目

uniqueUdpDstPortCnt()

：产生警报的Udp目的端口数

UDPPktCnt()

:与Tcp相关的报警数

UDPPktCnt()

:与Udp相关的报警数

PortscanPktCnt()

:与端口扫描相关的报警数

graph\_data\_class.php

class graph\_data

作用：用于产生general\_display.php文件需要的显示数据

变量：

$DB;

函数：

graph\_data($DB)

:构造函数

getTimeDataSet(&$xdata, $chart\_type, $time\_start, $time\_end)

:返回时间-警报数的二维数组的个数

$xdata保存改二维数组

$chart\_type图表类型

$time\_start时间的开始

$time\_end时间段的结束

getIpDataSet(&$xdata, $chart\_type, $time\_start, $time\_end)

：返回ip-警报数的二维数组的个数

$xdata保存改二维数组

$chart\_type图表类型

$time\_start时间的开始

$time\_end时间段的结束

getPortDataSet(&$xdata, $chart\_type, $time\_start, $time\_end)

:返回port-警报数的二维数组的个数

$xdata保存改二维数组

$chart\_type图表类型

$time\_start时间的开始

$time\_end时间段的结束

getSensorDataSet(&$xdata, $chart\_type, $time\_start, $time\_end)

：返回sensor-警报数的二维数组的个数

$xdata保存改二维数组

$chart\_type图表类型

$time\_start时间的开始

$time\_end时间段的结束

agLong2IP($long\_IP)

:将长整形的ip地址转换成点分制的ip地址

getSignatureName($sig\_id, $db)

:返回警报类型的名称

checkTime($time\_group)

:检查时间是否为空，及时间格式是否正确

StoreAlertNum($sql, $label, &$xdata, &$cnt, $min\_threshold)

:将时间-报警数存为二维数组形式

$sql执行的sql语句

$label x轴坐标显示的标签

$xdata 返回的二维数组

$cnt 二维数组的个数

$min\_threshold x轴的最小间距

approto\_class.php

class approto

作用：显示L7协议相关信息

变量

$DB

函数

getProtoDataSet(&$xdata)

:数目-Proto二维数组形式

:$xdata返回二维数组

返回值为二维数组的个数

system\_info\_class.php

class system\_info

作用：显示系统的信息

函数

system\_info()

:构造函数

GetkernelVersion()

:系统的内核版本

GetUptime()

:更新时间

interfaces目录下

admin目录下

管理员操作相关的文件

manage\_user.php

add\_user.php

管理用户，查看、添加、删除、修改用户信息。

manage\_rules.php

add\_rule.php

管理规则，查看、添加、删除、修个规则

info\_system.php

显示相关的系统信息

common目录下

general\_info.php

显示主页的报警信息

general\_display.php

主页图表显示tcp,udp,icmp,portscan的报警信息

view\_logs.php 日志查看文件与下面5个相关

graph\_main.php

graph\_form.php

graph\_display\_bar.php

graph\_display\_line.php

graph\_display\_pie.php

日志查看中的图表显示

graph\_main.php 主的逻辑控制页面，得到由graph\_form.php传递的参数然后发送参数给graph\_display\_\*.php不同文件来显示不同的图形

view\_proto.php L7协议信息参考相关的文件

proto\_display.php

控制图形的显示，如果没有数据，将不进行图形显示。

proto\_dispaly\_pie.php 饼图

目前view\_proto只能显示不同协议所占得比例，用饼图来表示。

user目录下

普通用户操作相关

manage\_rules.php

add\_rule.php

普通用户管理规则，查看、添加、删除、修个规则

lang目录下

ch-dictionary 中文语言

en-dictionary 英文语言