概要设计

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块名称** | 应用层协议分析模块 | | **父模块** | 无 | **标识** |  |
| **接口说明** | **输入** | 以太网数据包 | | | | |
| **输出** | 源IP、源端口、协议名、时间 | | | | |
| **功能说明** | 该模块用于对一个数据包的应用层协议做分析，来发现当前网络中使用的常用软件。首先，系统读取由用户指定的待检测协议，将用于协议匹配的正则表达式加载入内存。然后，当数据包到达时，系统将数据包中应用层数据匹配正则表达式，如果匹配成功，则输出源IP、源端口、匹配的时间、协议名到数据库 | | | | | |
| **支持环境** | Mysql：分析出的结果通过mysql输出  Snort：snort提供的以太网数据包作为输入 | | | | | |
| **依赖**  **关系** | **调用模块** | 规则文件管理模块  连接管理模块  规则匹配模块  输出过滤模块  输出模块 | | | | |
| **被调用模块** | Snort检测模块 | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块名称** | 规则文件管理模块 | | **父模块** | 应用层协议分析模块 | **标识** |  |
| **接口说明** | **输入** | 本地规则配置文件、WEB端规则配置文件 | | | | |
| **输出** | 规则正则表达式链表 | | | | |
| **功能说明** | 1． 读取本地规则配置文件。  2． 根据配置文件查找指定协议的规则文件。  3． 将规则文件中的正则表达式编译并加载入内存  4． 读取WEB端规则配置文件  5． 根据配置文件查找指定协议的规则文件。  6． 将规则文件中的正则表达式编译并加载入内存 | | | | | |
| **支持环境** | 本地规则配置文件：由管理员在本地指定的需要检测的应用层协议名  WEB端规则配置文件：由管理员在WEB端指定的需要检测的应用层协议名  本地规则文件夹：由管理员在本地指定的需要检测的规则文件集  WEB端规则文件夹：由管理员在WEB端指定的需要检测的规则文件集，其通过数据库来获取 | | | | | |
| **依赖**  **关系** | **调用模块** | 无 | | | | |
| **被调用模块** | 应用层协议分析模块 | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块名称** | 连接管理模块 | | **父模块** | 应用层协议分析模块 | **标识** |  |
| **接口说明** | **输入** | 以太网数据包 | | | | |
| **输出** | 源IP、源端口、目的IP、目的端口、协议名、协议号、时间 | | | | |
| **功能说明** | 1． 查找当前连接（由源IP,源端口号,目的IP,目的端口号组成）是否是旧连接，若是，则读取旧连接，否则新建一个连接。  2． 查看该连接是否已经匹配成功,若匹配成功,则跳过后续操作  3． 若该连接上数据未匹配成功过,则遍历正则表达式链表，匹配当前连接上应用层的数据  4． 若匹配成功，则输出源IP、源端口号、协议名、匹配成功的时间  5． 若连续匹配失败超过5次，则删除该连接 | | | | | |
| **支持环境** | Snort：snort提供的以太网数据包作为输入 | | | | | |
| **依赖**  **关系** | **调用模块** | 规则匹配模块 | | | | |
| **被调用模块** | 应用层协议分析模块 | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块名称** | 规则匹配模块 | | **父模块** | 连接管理模块 | **标识** |  |
| **接口说明** | **输入** | 经过连接过滤的以太网数据包 | | | | |
| **输出** | 分析出的协议号、协议名 | | | | |
| **功能说明** | 1． 去除应用层数据中的0x00，防止匹配不完整  2． 遍历正则表达式链表，逐个匹配正则表达式  3． 若匹配成功，则返回协议号和协议名称  4． 若匹配不成功，返回协议号为0 | | | | | |
| **支持环境** | 无 | | | | | |
| **依赖**  **关系** | **调用模块** | 无 | | | | |
| **被调用模块** | 连接管理模块 | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块名称** | 输出过滤模块 | | **父模块** | 应用层协议分析模块 | **标识** |  |
| **接口说明** | **输入** | 由规则匹配模块输出的协议号、协议名  由连接管理模块输出的源IP | | | | |
| **输出** | 过滤的结果（源IP，源端口、协议名、匹配时间） | | | | |
| **功能说明** | 1. 检查当前检测到的协议是否已经检查过 2. 若检查过，并且超过了检查间隙时间，则更新检查时间；反之，跳过后续操作 3. 若没检查过，则将记录下当前协议名和匹配时间等数据 | | | | | |
| **支持环境** | 无 | | | | | |
| **依赖**  **关系** | **调用模块** | 无 | | | | |
| **被调用模块** | 应用层协议分析模块 | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块名称** | 输出模块 | | **父模块** | 应用层协议分析模块 | **标识** |  |
| **接口说明** | **输入** | 输出过滤模块过滤后的匹配结果（源IP，源端口、协议名、匹配时间） | | | | |
| **输出** | 数据库操作语句字符串 | | | | |
| **功能说明** | 1. 写线程将数据拷贝到由写指针指定的缓冲区 2. 读线程将数据从由读指针指定的缓冲区读取数据，并执行数据库IO操作。 | | | | | |
| **支持环境** | Mysql：分析出的结果通过mysql输出 | | | | | |
| **依赖**  **关系** | **调用模块** | 无 | | | | |
| **被调用模块** | 应用层协议分析模块 | | | | |