(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 120429866 A
(43)申请公布日 2025.08.05

(21)申请号 202510514261.3

(22)申请日 2025.04.23

(71)申请人 北京火绒网络科技有限公司
地址 100000 北京市朝阳区北苑路30号院3号楼1至10层101号9层901

(72)发明人 周军 毛钧

(74)专利代理机构 北京博识智信专利代理事务所(普通合伙) 16067
专利代理师 时贝贝

(51)Int.Cl.
G06F 21/56(2013.01)
G06F 21/53(2013.01)
G06F 9/50(2006.01)

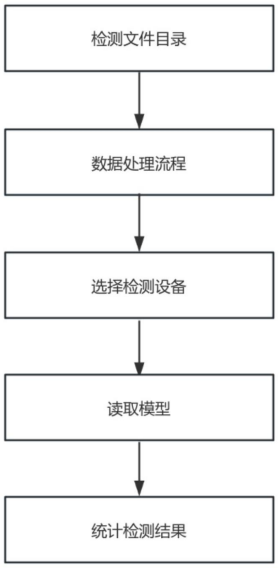
权利要求书2页 说明书5页 附图4页

(54)发明名称

一种基于NPU的恶意文件检测方法

(57)摘要

本发明涉及网络安全技术领域,尤其涉及一种基于NPU的恶意文件检测方法,包括以下步骤:步骤S1、数据处理:将火绒虚拟沙盒提取的动态行为日志作为原始输入数据进行处理;步骤S2、模型转化:将深度学习模型依次转化为ONNX模型和OpenVino IR模型;步骤S3、检测推理:将处理后的输入数据和OpenVino IR模型加载到NPU上进行模型推理,获得推理结果。本发明通过将火绒虚拟沙盒提取的动态行为日志预处理、分词并建立词表,再经Pytorch模型转化为OpenVino IR模型后在NPU上推理,实现了高效处理海量日志数据,降低硬件资源占用与功耗,提升模型推理效率,解决了传统方案在实时检测和边缘计算场景中能效不足的问题。



CN 120429866 A

1.一种基于NPU的恶意文件检测方法,其特征在于:包括以下步骤:

步骤S1、数据处理:将火绒虚拟沙盒提取的动态行为日志作为原始输入数据进行处理;

步骤S2、模型转化:将深度学习模型依次转化为ONNX模型和OpenVino IR模型;

步骤S3、检测推理:将处理后的输入数据和OpenVino IR模型加载到NPU上进行模型推理,获得推理结果。

2.根据权利要求1所述的一种基于NPU的恶意文件检测方法,其特征在于:在步骤S1中,所述数据处理包括以下步骤:

步骤S11、获得火绒虚拟沙盒动态行为日志:将文件放入火绒虚拟沙盒中,得到动态行为日志;

步骤S12、动态行为日志预处理,所述预处理包括删除无效API序列、删除API序列前后标点符号以及API序列连接;

步骤S13、使用pytorch自然语言处理库torchtext对动态行为日志进行分词:拆分API序列,统计tokens;

步骤S14、建立词表:建立词汇字典,用于获得任何输入API序列的token id。

3.根据权利要求2所述的一种基于NPU的恶意文件检测方法,其特征在于:在步骤S2中,所述模型转化包括以下步骤:

步骤S21、使用Pytorch深度学习框架训练神经网络模型model.pt;

步骤S22、使用Pytorch将model.pt转化为model.onnx格式的权重,作为中间过渡模型。

步骤S23、使用openvino框架转化ONNX模型为OpenVino IR模型。

4.根据权利要求3所述的一种基于NPU的恶意文件检测方法,其特征在于:在步骤S2中,所述Pytorch模型包括输入层、卷积层、池化层以及全连接层和softmax层;

所述输入层是一个n*k的矩阵,n表示句子中的单词数,k表示每个词对应的词向量的维度;所述输入层的每一行表示一个单词的k维词向量;

所述卷积层使用多个不同大小的卷积核对输入的词向量进行卷积操作;每个卷积核在句子上滑动,提取不同范围内的词的关系;

所述池化层采用1-Max池化,从每个滑动窗口产生的特征向量中筛选出一个最大的特征,并将特征拼接起来构成向量表示;

所述全连接层和softmax层将池化层的输出传递给全连接层,并使用Softmax激活函数输出每个类别的概率。

5.根据权利要求4所述的一种基于NPU的恶意文件检测方法,其特征在于:在步骤S23中,所述OpenVino使用IR文件格式作为神经网络数据处理格式。

6.根据权利要求5所述的一种基于NPU的恶意文件检测方法,其特征在于:在步骤S23中,IR模型包括XML文件以及BIN文件;所述XML文件描述网络拓扑结构,所述BIN文件包含网络权重数据以及IR文件格式存储。

7.根据权利要求6所述的一种基于NPU的恶意文件检测方法,其特征在于:在步骤S3中,所述检测推理包括以下步骤:

步骤S31、上传待检测文件目录;

步骤S32、对原始数据格式进行处理;

步骤S33、选择检测设备:自动识别当前机器可供使用的设备,并在NPU上进行推理;

步骤S34、读取模型:加载OpenVino IR模型,产生检测结果;

步骤S35、统计检测结果:对上传文件的检测结果进行统计汇总。

一种基于NPU的恶意文件检测方法

技术领域

[0001] 本发明涉及网络安全技术领域,尤其涉及一种基于NPU的恶意文件检测方法。

背景技术

[0002] 近年来,恶意文件检测技术经历了从传统方法向智能化方向的快速发展。早期检测手段主要依赖静态特征码匹配,通过比对已知恶意文件的特征数据库实现识别。然而,随着恶意软件变种的爆发式增长及混淆技术的广泛应用,基于签名的检测方法逐渐暴露出滞后性和高误报率的缺陷。为应对这一挑战,基于动态行为分析的沙盒技术应运而生,通过模拟执行环境捕获文件运行时的API调用序列、系统行为等动态特征,显著提升了未知恶意软件的识别能力。与此同时,深度学习技术的引入进一步推动了检测效能的突破,基于卷积神经网络等模型的行为特征提取与分类成为主流方案,模型通常在CPU或GPU上进行训练与推理,实现了更高维度的特征抽象与分类精度。

[0003] 然而,现有技术仍存在诸多局限性。传统沙盒技术在处理海量动态行为日志时,存在资源占用高、分析效率低的问题,难以满足实时检测需求;而基于CPU或GPU的深度学习推理方案虽提升了检测准确性,但面临硬件功耗高、算力利用率不足的瓶颈,尤其在边缘计算或低功耗场景下难以平衡性能与能效。此外,深度学习模型从训练框架到部署环境的转换过程复杂,涉及多阶段格式转化与硬件适配,导致部署周期延长、灵活性下降。如何在高动态、低延时的实际场景中实现高效、低耗的恶意文件检测,仍是亟待解决的技术难题。

发明内容

[0004] 本发明的目的在于克服上述问题,提供一种基于NPU的恶意文件检测方法,为实现上述目的,本发明采用如下技术方案:

[0005] 一种基于NPU的恶意文件检测方法,包括以下步骤:

[0006] 步骤S1、数据处理:将火绒虚拟沙盒提取的动态行为日志作为原始输入数据进行处理;

[0007] 步骤S2、模型转化:将深度学习模型依次转化为ONNX模型和OpenVinoIR模型;

[0008] 步骤S3、检测推理:将处理后的输入数据和OpenVinoIR模型加载到NPU上进行模型推理,获得推理结果。

[0009] 进一步地,在步骤S1中,所述数据处理包括以下步骤:

[0010] 步骤S11、获得火绒虚拟沙盒动态行为日志:将文件放入火绒虚拟沙盒中,得到动态行为日志;

[0011] 步骤S12、动态行为日志预处理,所述预处理包括删除无效API序列、删除API序列前后标点符号以及API序列连接;

[0012] 步骤S13、使用pytorch自然语言处理库torchtext对动态行为日志进行分词:拆分API序列,统计tokens;

[0013] 步骤S14、建立词表:建立词汇字典,用于获得任何输入API序列的tokenid。

[0014] 进一步地,在步骤S2中,所述模型转化包括以下步骤:

[0015] 步骤S21、使用Pytorch深度学习框架训练神经网络模型model.pt;

[0016] 步骤S22、使用Pytorch将model.pt转化为model.onnx格式的权重,作为中间过渡模型。

[0017] 步骤S23、使用openvino框架转化ONNX模型为OpenVino IR模型。

[0018] 进一步地,在步骤S2中,所述Pytorch模型包括输入层、卷积层、池化层以及全连接层和softmax层;

[0019] 所述输入层是一个n*k的矩阵,n表示句子中的单词数,k表示每个词对应的词向量的维度;所述输入层的每一行表示一个单词的k维词向量;

[0020] 所述卷积层使用多个不同大小的卷积核对输入的词向量进行卷积操作;每个卷积核在句子上滑动,提取不同范围内的词的关系;

[0021] 所述池化层采用1-Max池化,从每个滑动窗口产生的特征向量中筛选出一个最大的特征,并将特征拼接起来构成向量表示;

[0022] 所述全连接层和softmax层将池化层的输出传递给全连接层,并使用Softmax激活函数输出每个类别的概率。

[0023] 进一步地,在步骤S23中,所述OpenVino使用IR文件格式作为神经网络数据处理格式。

[0024] 进一步地,步骤S23中,IR模型包括XML文件以及BIN文件;所述XML文件描述网络拓扑结构,所述BIN文件包含网络权重数据以及IR文件格式存储。

[0025] 进一步地,在步骤S3中,所述检测推理包括以下步骤:

[0026] 步骤S31、上传待检测文件目录;

[0027] 步骤S32、对原始数据格式进行处理;

[0028] 步骤S33、选择检测设备:自动识别当前机器可供使用的设备,并在NPU上进行推理;

[0029] 步骤S34、读取模型:加载OpenVino IR模型,产生检测结果;

[0030] 步骤S35、统计检测结果:对上传文件的检测结果进行统计汇总。

[0031] 本发明的优点在于:

[0032] 1、本发明通过将火绒虚拟沙盒提取的动态行为日志进行预处理、分词及建立词表等数据处理,再把深度学习模型依次转化为ONNX模型和OpenVino IR模型,最后在NPU上进行推理,实现了在处理动态行为日志时降低资源占用、提升分析效率,且利用NPU低功耗特性,在边缘计算或低功耗场景中平衡性能与能效,相比CPU和GPU降低了功耗。

[0033] 2、本发明通过在检测推理时自动识别当前机器可用设备并默认在NPU上推理,加载OpenVino IR模型进行检测并统计结果,实现了利用NPU提升算力利用率,解决了传统基于CPU或GPU推理方案硬件功耗高、算力利用率不足的问题,能在高动态、低延时场景中实现高效的恶意文件检测。

附图说明

[0034] 构成本申请的一部分的附图用来提供对本申请的进一步理解,使得本申请的其它特征、目的和优点变得更明显。本申请的示意性实施例附图及其说明用于解释本申请,并不

构成对本申请的不当限定。

[0035] 在附图中:

[0036] 图1为实施例1中一种基于NPU的恶意文件检测方法的数据处理流程图。

[0037] 图2为实施例1中一种基于NPU的恶意文件检测方法的部分行为日志。

[0038] 图3为实施例1中一种基于NPU的恶意文件检测方法的模型转化流程图。

[0039] 图4为实施例1中一种基于NPU的恶意文件检测方法的检测流程图。

[0040] 图5为实施例1中一种基于NPU的恶意文件检测方法的启动界面。

[0041] 图6为实施例1中一种基于NPU的恶意文件检测方法的扫描完成界面。

[0042] 图7为实施例1中一种基于NPU的恶意文件检测方法的数据统计界面。

具体实施方式

[0043] 为使本发明实施例的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。通常在此处附图中描述和示出的本发明实施例的组件可以以各种不同的配置来布置和设计。

[0044] 下面通过具体实施例对本发明进行详细和具体的介绍,以使更好的理解本发明,但是下述实施例并不限定本发明的保护范围。

[0045] 实施例1

[0046] 如图1-7所示,一种基于NPU的恶意文件检测方法,包括以下步骤:

[0047] 步骤S1、数据处理:将火绒虚拟沙盒提取的动态行为日志作为原始输入数据进行处理;

[0048] 步骤S2、模型转化:将深度学习模型依次转化为ONNX模型和OpenVino IR模型;

[0049] 步骤S3、检测推理:将处理后的输入数据和OpenVino IR模型加载到NPU上进行模型推理,获得推理结果。

[0050] 本方法包含数据处理、模型转化、检测推理三个核心步骤。数据处理环节以火绒虚拟沙盒提取的动态行为日志为原始输入数据并进行处理;模型转化环节将深度学习模型依次转化为ONNX模型和OpenVino IR模型;检测推理环节把处理后的输入数据和OpenVino IR模型加载到NPU上开展模型推理,从而获得推理结果,以此实现基于NPU的恶意文件检测。

[0051] 进一步地,在步骤S1中,所述数据处理包括以下步骤:

[0052] 步骤S11、获得火绒虚拟沙盒动态行为日志:将文件放入火绒虚拟沙盒中,得到动态行为日志;

[0053] 步骤S12、动态行为日志预处理,所述预处理包括删除无效API序列、删除API序列前后标点符号以及API序列连接;

[0054] 步骤S13、使用pytorch自然语言处理库torchtext对动态行为日志进行分词:拆分API序列,统计tokens;

[0055] 步骤S14、建立词表:建立词汇字典,用于获得任何输入API序列的token id。

[0056] 例如:数据转化过程

[0057] 1、把api序列转成列表;

[0058] ['crtdll:__GetMainArgs','KERNEL32:GetModuleFileNameA','KERNEL32:

GetCommandLineA']；

[0059] 2、使用词模型把api列表,转成数值列表,[987,15,23]；

[0060] 3、把数值列表转出tensor格式,tensor([987,15,23])；

[0061] 4、填充tensor到指定长度,长度不够的补0,tensor([987,15,23,0,0])；

[0062] 5、增加tensor维度,tensor([[987,15,23,0,0]])。

[0063] 在数据处理步骤中,首先将文件放入火绒虚拟沙盒,获取包含API序列等信息的动态行为日志;接着对日志进行预处理,删除无效API序列、前后标点符号,并进行API序列连接,净化数据;然后借助pytorch的torchtext库拆分API序列、统计tokens来完成分词;最后建立词汇字典即词表,用于将输入的API序列转换为对应的token id,为后续模型处理准备标准化数据。

[0064] 部分行为日志如图2所示,数据样本示例如下：

[0065] Sha1:0e63ef0822ec086378f5d953792629c1eec24c32

[0066] 进一步地,在步骤S2中,所述模型转化包括以下步骤：

[0067] 步骤S21、使用Pytorch深度学习框架训练神经网络模型model.pt；

[0068] 步骤S22、使用Pytorch将model.pt转化为model.onnx格式的权重,作为中间过渡模型。

[0069] 步骤S23、使用openvino框架转化ONNX模型为OpenVino IR模型。

[0070] 模型转化步骤中,先使用Pytorch框架训练得到神经网络模型model.pt;再利用Pytorch将其转化为model.onnx格式的权重文件,作为中间过渡模型,该文件包含网络流动、输入输出等信息;最后通过openvino框架将ONNX模型转化为OpenVino IR模型,以便适配NPU硬件进行部署。

[0071] 进一步地,在步骤S2中,所述Pytorch模型包括输入层、卷积层、池化层以及全连接层和softmax层；

[0072] 所述输入层是一个n*k的矩阵,n表示句子中的单词数,k表示每个词对应的词向量的维度;所述输入层的每一行表示一个单词的k维词向量；

[0073] 所述卷积层使用多个不同大小的卷积核对输入的词向量进行卷积操作;每个卷积核在句子上滑动,提取不同范围内的词的关系；

[0074] 所述池化层采用1-Max池化,从每个滑动窗口产生的特征向量中筛选出一个最大的特征,并将特征拼接起来构成向量表示；

[0075] 所述全连接层和softmax层将池化层的输出传递给全连接层,并使用Softmax激活函数输出每个类别的概率。

[0076] Pytorch模型包含输入层、卷积层、池化层、全连接层和softmax层。输入层是n*k矩阵,每行表示单词的k维词向量;卷积层运用不同大小卷积核对词向量卷积,滑动提取不同范围词的关系;池化层采用1-Max池化,从滑动窗口特征向量中筛选最大特征并拼接;全连接层和softmax层将池化输出处理后,通过Softmax函数输出每个类别的概率,实现对恶意文件的分类检测。

[0077] 进一步地,在步骤S23中,所述OpenVino使用IR文件格式作为神经网络数据处理格式。

[0078] 在将ONNX模型转化为OpenVino IR模型的过程中,OpenVino采用IR文件格式作为

神经网络数据处理格式,该格式为模型在NPU上的高效推理提供了统一且适配的处理规范,确保模型结构和数据能够被NPU正确识别和处理。

[0079] model.onnx代表ONNX格式的权重文件,这个权重文件不仅包含了权重值,也包含了神经网络的网络流动信息以及每一层网络的输入输出信息和一些其他的辅助信息。

[0080] 进一步地,步骤S23中,IR模型包括XML文件以及BIN文件;所述XML文件描述网络拓扑结构,所述BIN文件包含网络权重数据以及IR文件格式存储。

[0081] OpenVino IR模型由XML文件和BIN文件组成,其中XML文件用于描述神经网络的拓扑结构,清晰呈现网络各层的连接和组织方式;BIN文件则包含网络权重数据,并以IR文件格式存储,二者共同构成完整的模型文件,为NPU加载和执行模型推理提供必要的结构和数据支持。

[0082] 进一步地,在步骤S3中,所述检测推理包括以下步骤:

[0083] 步骤S31、上传待检测文件目录;

[0084] 步骤S32、对原始数据格式进行处理;

[0085] 步骤S33、选择检测设备:自动识别当前机器可供使用的设备,并在NPU上进行推理;

[0086] 步骤S34、读取模型:加载OpenVino IR模型,产生检测结果;

[0087] 步骤S35、统计检测结果:对上传文件的检测结果进行统计汇总。

[0088] 检测推理环节,先上传待检测文件目录,然后处理原始数据格式使其适配模型;接着自动识别可用设备并选择在NPU上推理,发挥NPU优势;再加载OpenVino IR模型进行检测并产生结果;最后统计汇总上传文件的检测结果,形成完整的检测报告,实现对恶意文件的实际检测应用。

[0089] 以上对本发明的具体实施例进行了详细描述,但其只是作为范例,本发明并不等同于以上描述的具体实施例。对于本领域技术人员而言,任何对本发明进行的等同修改和替代也都在本发明的范畴之中。因此,不脱离本发明的精神和范围下所做的均等变换和修改,都应涵盖在本发明的范围内。

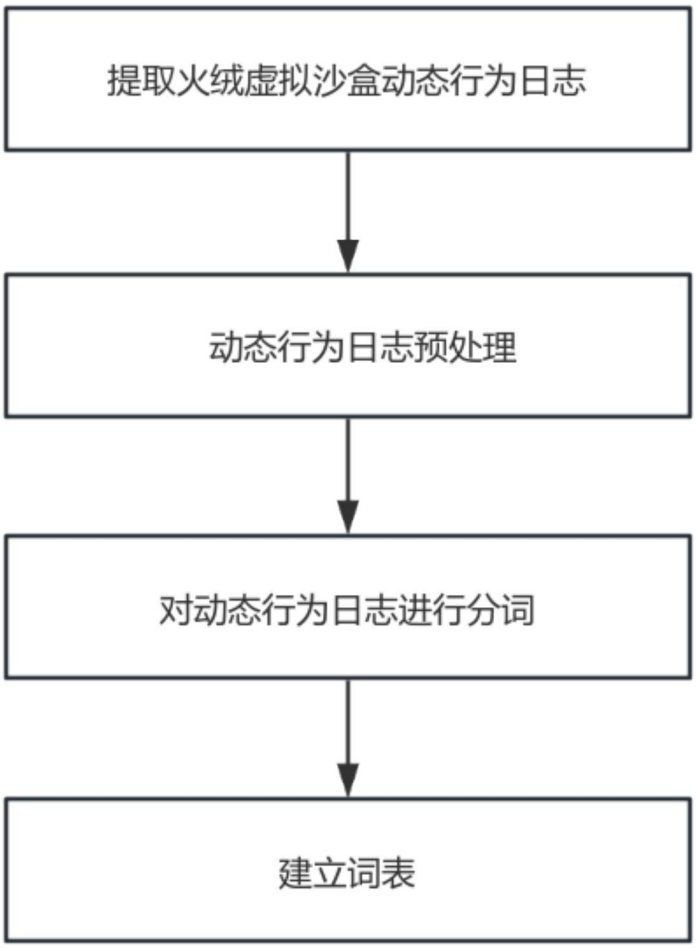


图1

	A
1	KERNEL32:GetCurrentThreadId:
2	KERNEL32:LoadLibraryExW:
3	KERNEL32:GetProcAddress:
4	KERNEL32:InitializeCriticalSectionEx:
5	KERNEL32:LoadLibraryExW:
6	KERNEL32:LoadLibraryExW:
7	KERNEL32:GetProcAddress:
8	KERNEL32:GetProcAddress:
9	KERNEL32:LoadLibraryExW:
10	KERNEL32:GetProcAddress:

图2

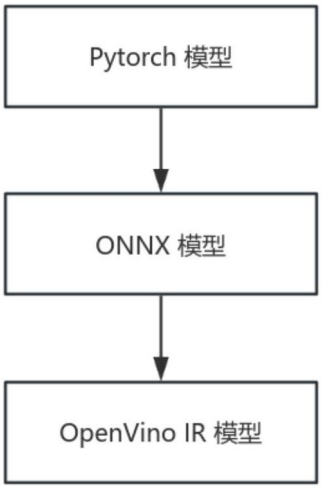

Pytorch 模型
ONNX 模型
OpenVino IR 模型

图3

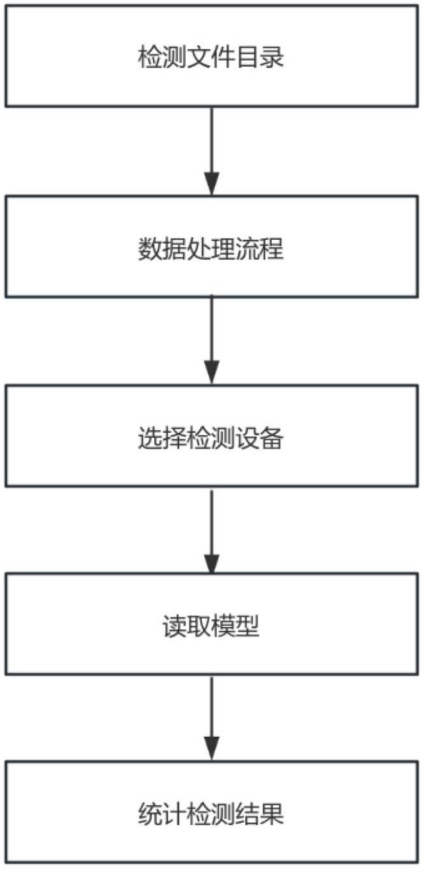

检测文件目录
数据处理流程
选择检测设备
读取模型
统计检测结果

图4


火绒安全
自定义扫描，查杀木马病毒
基于AI的威胁检测与分析以及混合架构下的XPU性能优化
自定义扫描
使用以下设备进行推理：
CPU GPU NPU
检测到NPU，默认使用NPU进行AI计算
直接拖拽目标文件到此窗口将自动开始扫描

图5

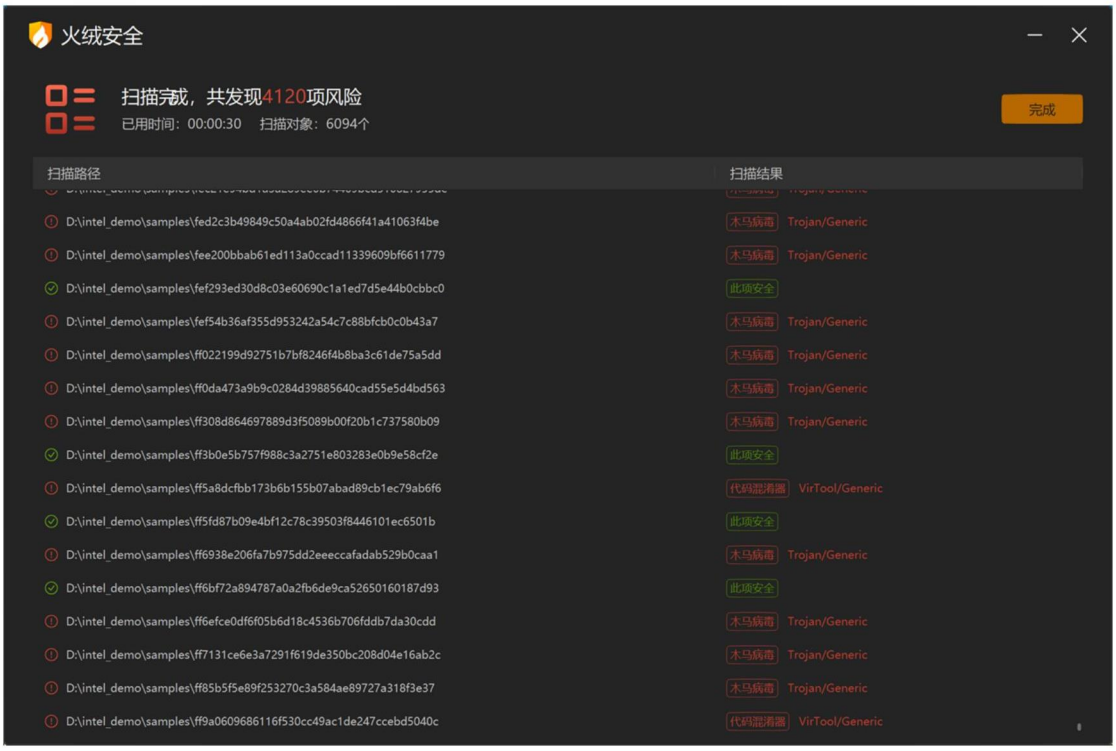

火绒安全
扫描完成，共发现4120项风险
已用时间：00:00:30 扫描对象：6094个
完成
扫描路径
扫描结果
D:\intel_demo\samples\fed2c3b49849c50a4ab02fd4866f41a41063f4be 木马病毒 Trojan/Generic
D:\intel_demo\samples\fee200bbab61ed113a0ccad11339609bf6611779 木马病毒 Trojan/Generic
D:\intel_demo\samples\fef293ed30d8c03e60690c1a1ed7d5e44b0cbbc0 此项安全
D:\intel_demo\samples\fef54b36af355d953242a54c7c88bfcb0c0b43a7 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff022199d92751b7bf8246f4b8ba3c61de75a5dd 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff0da473a9b9c0284d39885640cad55e5d4bd563 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff308d864697889d3f5089b00f20b1c737580b09 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff3b0e5b757f988c3a2751e803283e0b9e58cf2e 此项安全
D:\intel_demo\samples\ff5a8dcfbb173b6b155b07abad89cb1ec79ab6f6 代码混淆器 VirTool/Generic
D:\intel_demo\samples\ff5fd87b09e4bf12c78c39503f8446101ec6501b 此项安全
D:\intel_demo\samples\ff6938e206fa7b975dd2eeeccafadab529b0caa1 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff6bf72a894787a0a2fb6de9ca52650160187d93 此项安全
D:\intel_demo\samples\ff6efce0df6f05b6d18c4536b706fddb7da30cdd 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff7131ce6e3a7291f619de350bc208d04e16ab2c 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff85b5f5e89f253270c3a584ae89727a318f3e37 木马病毒 Trojan/Generic
D:\intel_demo\samples\ff9a0609686116f530cc49ac1de247ccebd5040c 代码混淆器 VirTool/Generic

图6


火绒安全
扫描对象
6094个
风险文件
4120个
无风险文件
1974个
扫描对象
风险文件 4120 个 68%
无风险文件 1974 个 32%
风险文件
68%
无风险文件
32%
病毒类型
TrojanDownloader 90
TrojanDropper
TrojanClicker
TrojanSpy 82
Backdoor 58
Ransom
Adware 17
VirTool 794
Rogue
Worm 1
Trojan 3074
Virus 4

图7