**“野外”计算机安全漏洞数据量化的初步分析**

**摘要**

一个由计算机、网络和软件组成的系统存在一定程度的漏洞，其面临黑客的犯罪风险。目前，大多数漏洞研究使用软件供应商的数据和国家漏洞数据库（NVD）。我们提出了另一条前进的道路，将我们的分析建立在操作信息安全社区的数据基础上，即“野外”的脆弱性数据。本文提出了一种脆弱性数据解析算法，并对漏洞来临和删除过程（又称漏洞出生-死亡过程）进行了单变量和多变量深入分析。我们发现，漏洞到达的最显著特征是对数正态分布，而漏洞删除的最显著特征是指数分布。这些分布可以作为未来贝叶斯分析的先验概率。我们还发现，超过22%的已删除漏洞数据的速率为零，并且到达漏洞数据总是大于零。最后，我们通过二变量散点图和统计观察来量化和可视化漏洞到达和删除之间的依赖性。

**关键词：脆弱性数据；操作脆弱性管理；NESSUS扫描仪；探索性数据分析；重尾分布；单变量分析；双变量分析INTRODUCTION**

**1.导论**

信息安全从业人员在修补网络漏洞方面面临着双重挑战。它们的补丁数量超出了它们的响应能力。网络渗透测试软件输出大量数据，包括常见漏洞枚举（CVE）、常见漏洞评分系统（CVSS）、简要描述和漏洞修补信息。事实上，美国政府和行业的最佳做法是将漏洞风险与CVSS基准分数相关联。然而，越来越多的证据表明，计算CVSS值的方法可能需要重新评估。

为了使漏洞缓解变得更具挑战性，安全工程师在严格的限制下工作。他们的时间、资源和人力有限。在理想情况下，他们会做出有针对性和精确的决定，首先对最关键的机器进行补丁。然而，外部因素，例如一台机器对另一台机器的价值，在当前的脆弱性软件风险度量中被完全忽略。因此，系统管理员必须在内部启动项目，并将企业级网络划分为不同关键级别的区域。然后，他们必须将这些区域与漏洞输出进行匹配，以确定应首先解决哪些机器的向下选择过程，形势严峻，许多因素阻碍了管理IT环境中真正具有攻击性的漏洞。

本文对“野外”的脆弱性数据进行了初步分析。我们的最终目标是促进解决方案，以解决漏洞管理和数据洞察方面的当前IT挑战。我们通过进行探索性数据分析和量化脆弱性出生和死亡过程之间的关系来评估操作IT脆弱性暴露的状态，从而实现这一点。在第1.1节中，我们回顾了我们的目标研究成果，并概述了论文的其余部分。

我们在研究中解决了上述问题的多个方面，即：

1. 开发数据解析算法，简化漏洞数据处理。
2. 统计特征化漏洞到达和删除过程。
3. 清楚地描述漏洞到达和删除之间存在哪些依赖关系。
4. 量化单变量和双变量关系，并解释其在“野外”总脆弱性暴露方面的含义。
5. 为我们的研究确定对内部和外部有效性的威胁。
6. 总结和概述我们研究的未来工作。

我们认为，上述成果对于更深入地了解当前的操作漏洞管理状况和改进最新技术的机会领域至关重要。

在第2节中，我们对脆弱性分析中的相关工作进行了全面的概述。此外，由于本研究侧重于评估使组织面临犯罪剥削风险的整体脆弱性暴露，因此我们还为感兴趣的读者提供了不同风险模型和指标的概述。在第3、4和5节中，我们描述了数据、分析数据的方法和结果。在第6节和第7节中，我们将讨论未来工作的机会领域以及我们的结论。

2.相关工作

在本节中，我们概述了有关漏洞模型、风险评估和影响漏洞管理的框架的相关工作。

建模和统计理解漏洞可以为漏洞管理和缓解过程提供更深入的见解。最近，已经提出了两个评估漏洞暴露的指标，即：1）活动漏洞中位数（MAV）；2）无漏洞天数（VFD）。1这些指标是根据脆弱性生命线模型推导出来的，该模型基于测量何时向供应商报告漏洞，以及何时向同一供应商报告漏洞。UE补丁。我们的工作是不同的，因为我们专门研究操作IT数据。基于供应商何时发现新的漏洞的分析，当他们发布补丁时，这为建模漏洞提供了坚实的第一步；但是，它并不能完全捕获漏洞补丁“现实世界”部署的成功和失败。仅仅因为知道了一个漏洞，或者供应商提供了一个补丁，并不一定意味着补丁被有效地使用了。一种解决方案是将来自真实漏洞扫描和企业级渗透测试的现场数据结合起来，以获得更全面的理解。

基于通用脆弱性评分系统（CVSS），对脆弱性管理进行了研究。CVSS度量提供了分配漏洞严重性的基本分数。不同的级别由卡内基甜瓜大学的CERT、软件供应商和NIST等组织定义。漏洞通常由CVSS基准值进行分类，该基准值不包括时间或环境因素。最近的工作之一是将黑市利用数据等外部因素纳入CVSS基本得分。其他工作还对基于CVSS数据的18个安全评估指标进行了统计分析，并对34个成功攻击进行了时间折中。他们发现，仅使用CVSS数据进行安全建模并不能准确描述系统崩溃的时间。他们还发现，仅基于最严重的CVSS数据的安全模型不如考虑所有漏洞的安全模型可靠，无论其CVSS严重性如何。

相关工作的另一个领域来源于计算和网络风险模型。具体来说，Clark等人通过任务树将一个组织的目标和关系与网络主机联系起来，以模拟风险。开发了4个其他风险模型，以在考虑到漏洞之前考虑到需要保护的网络资产。5此外，对投资效率的风险进行定量评估NT策略（查询）为使用新兴技术进行网络安全风险评估提供了一种强大的跨学科方法。它提供了一种计算方法，用于导航软件和知识产权（IP）等示例中保护最具成本效益的策略。.

其他工作可以通过微软的威胁建模方法找到，这是一种在整个系统开发生命周期中建模网络威胁的方法。它根据Stride对威胁进行分类，Stride代表欺骗、篡改、拒绝、信息披露、拒绝服务、提升权限。该方法跟踪相关的攻击技术、目标和可以执行的补救措施。

最后，其他三个评估风险的模型是塔拉、莫达和达卡提供了评估风险的方法。tara是威胁评估和修复分析的缩写。它是一种用于应对网络威胁的系统级方法，并使用评分模型和威胁矩阵等工具。它可以在已经部署的系统以及仍在采办生命周期中的系统上执行。它的优势在于，通过调整最适合系统的评估级别，它提供了一种灵活的方法。此外，只需使用给定系统所需的工具。MORDA是面向任务的风险和设计分析的缩写。它是系统级的风险评估框架。它利用了对手、用户和服务提供者的模型。目标是最大化用户价值，无论系统是否受到攻击。这些模型依赖于来自主题专家（SME）的信息，并建立在已知的对手信息和攻击模式的基础上。最后，DACCA是对抗网络攻击的决策分析的缩写。DACCA对攻击影响进行权衡，并对该影响作出可能的响应。它采用主题专家（SME）评估来告知攻击的严重性、攻击的可能性以及对手能力的复杂性。

另一个有趣的方法是由麻省理工学院林肯实验室生产的netspa。它是一个映射计算机网络的软件，最容易受到黑客攻击。它以网络配置信息为输入，输出最坏情况下的场景攻击图。最终状态需要确定为一个目标，以及哪些主机可以被信任，最后是最大递归深度优先搜索大小。

在本节中，我们概述了相关工作的许多不同方面，从脆弱性模型到网络风险管理。在下一节中，我们将回顾我们的数据以及从原始数据到我们分析的最终数据所采用的“清理”过程。

3.数据

在本节中，我们将描述用于分析的数据，以及数据源和软件。在第4节中，我们概述了我们的方法和对数据进行预处理所遵循的步骤。

3.1数据描述

我们分析了Nessus漏洞扫描器的输出，以进行研究。第3.2节描述了Nessus的特定软件信息。我们研究了2013-2014两年期间的17组扫描数据。数据包含近3000个主机，11个不同的主机。图1显示了原始漏洞数据的示例。

Pugin lD

CVE

LV55

9

Nowt

192.168.150.131

Protocol

ten

Port

Name

oFTPD File

Newline

21

Character

Overflow

rlopsia

Don of

Prbn

The remote oast Is runnings version of PrordoO woad seems to be

vulnerable to a buffer overflow when a user downloads a malformed Alin

file.

An attazker with upload privileges on this host may abuse this flaw to gaina

rootsbell on

this bote.

'

The author of ProFTPD did not Increase the version number

of nu Product when fixing this Issue, sic It might be false

Solution

pg

eta ProFTP01.2.9

abed available hrMl.3.gp

Puugin Output

11809

CVE.20030831

bi[rarywtle may

...sea

remote

sew.

35777

LVg-19990502

10

192.168.150.13

Default

22

PaxwaM

'rood Account

Echo Service

Sul a

an imi

remote host uosa

known Pnk...r. .

ri echo sercirv

A

ThexwuntIra..onthe remote hast has the password Ito, M attacker

may leverage Mis issue to gain total coMml of the affected system.

The

ne

running

the.. service.

This

senate ecMez anytlata

phi. is sent to

remote host

This service is unused these days,. It

strongly assises

that you diteb,oit, as it may be used by attackers b

set

up Genial

as...

.

attxkz againztthis host.

chhh.t the

passwoM

Mis azcaunt oralsablo it

Awes possible. executethe aammana it on the re mote dost

wawlrohq gidDlrootl grop6S=D5aat7

LYE - 19990103

.2.168.150.13

ap

Under Onix systems comment omthe echti line in /eto/Inetdco

-

ana restart the meta process

Under W.W.I systems, set the following registry .y te 0:

IAt\ lyuem\ CUeemfnmrd5etüemses .insp7Mparameters \Fn

Ng

lq6c

flgtM\ lystemK ument!north,stt\lervires sihht.ct ParameteraEhh

bleildpEsto Then launch emd.eae and

type:

netstop slmptep

net start ssptcp To restart the service.

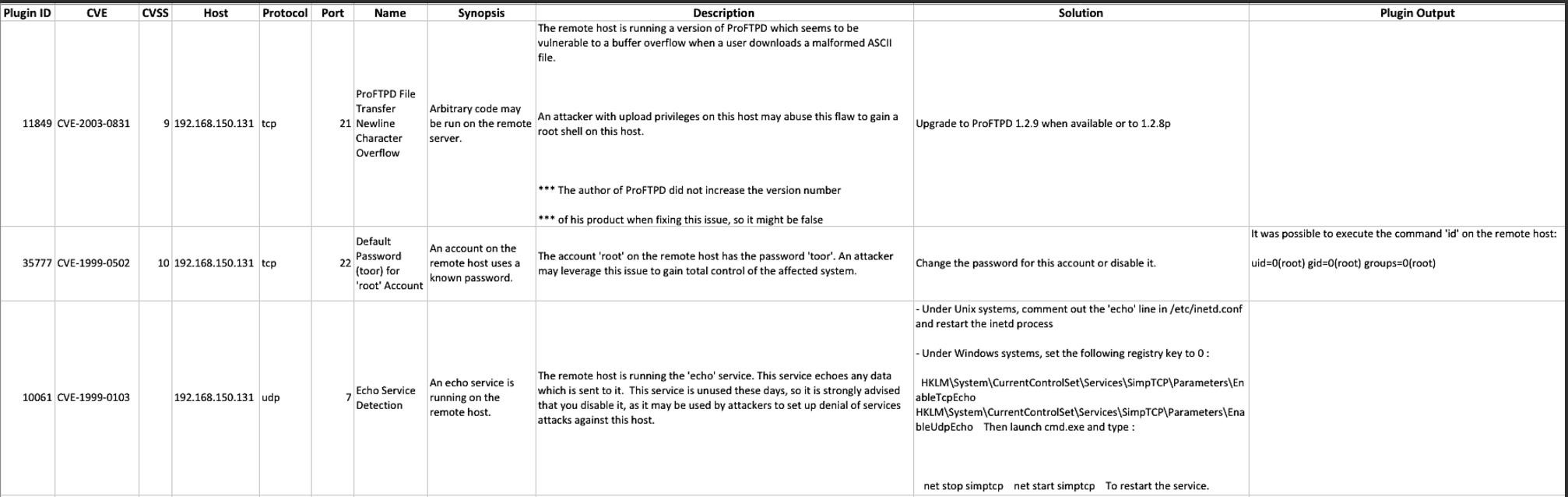


图1：典型Nessus漏洞输出文件的示例。

1. 插件ID：一个软件代码，每个代码都有一个唯一的ID号，用来识别它正在扫描的属性和漏洞。
2. 常见漏洞枚举（CVE）代码：用唯一可识别号码标记最常见漏洞的系统。
3. 通用漏洞评分系统（CVSS）：如第2节所述，CVSS值提供了分配漏洞严重性的基本分数。
4. 风险等级：用一个词描述漏洞的严重性。软件供应商通常使用“关键”、“高”、“中”、“低”或“信息”等描述符。CVSS值通知大多数软件供应商的风险级别描述。
5. 主机名/IP地址：主机名是标识连接到网络的机器或设备的标签，例如www.dartmouth.edu。IP地址是唯一标识连接到网络的机器或设备的数字字符串，例如172.16.254.1。任何机器或设备都可以同时具有IP地址和主机名，或者仅具有IP地址。.
6. 协议：指定主机、机器或设备与另一主机、机器或设备进行“对话”的通信方法和规则集。TCP和UDP是协议的例子。
7. 端口号：主机、机器或设备上通信端点的标识符，由逻辑结构确定。
8. 漏洞名称：漏洞的简短标题。通常少于5个单词，由软件供应商选择。
9. 脆弱性概要：对脆弱性的一句简短解释。
10. 脆弱性描述：对脆弱性的扩展和更全面的解释。可以提供当前和背景信息。长度可以是20-100个字。
11. 解决方案：修补漏洞的建议。示例包括“禁用SSLv3”或“为此应用程序联系软件供应商”。

3.2数据源和软件

生成我们数据的软件是Nessus漏洞扫描程序，版本5.0.1。Nessus由Tenable Security Solutions开发和维护。11漏洞扫描在达特茅斯计算服务数据中心运行。数据中心被称为组织的“大脑”，它集中存储、管理和传播对组织的计算机和网络操作至关重要的数据。它可以作为虚拟服务或物理服务存在。.

3.3最终数据集

数据经过某些关键处理阶段，从原始数据源获取最终数据。在本节中，我们简要概述了最终的数据集，在第4节中，我们更详细地描述了数据解析过程。

表1显示了数据预处理的最终输出。请注意，这些值表示每个类别在时间上的总体速率。它们不代表时间间隔之间的速率，例如从扫描到扫描。作为未来工作，第6节将讨论分析时间间隔之间脆弱性数据的变化。

表1：最终漏洞数据示例。.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Host | Arrival Rate | Deletion Rate | Disappearance Rate | Reappearance Rate |
| A | 0.529411765 | 0.529411765 | 0.529411765 | 0 |
| B | 2.882352941 | 2.05882353 | 2.823529412 | 0.764705882 |
| C | 0.352941176 | 0.352941176 | 0.352941176 | 0 |
| D | 0.176470588 | 0 | 0 | 0 |
| E | 1.411764706 | 0.294117647 | 0.294117647 | 0 |

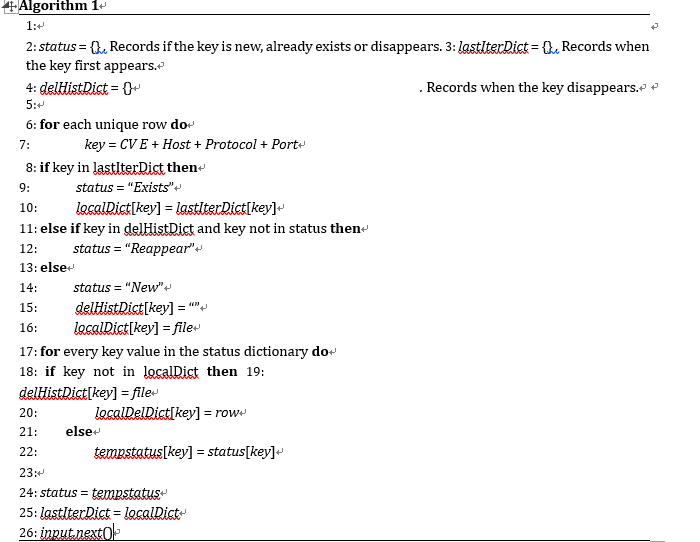
In the next section, we overview the methodology behind our data parsing algorithm and statistical analysis.

4.方法论

在本节中，我们将回顾数据的分析方法。在本研究中，我们分析了脆弱性到达过程和脆弱性删除过程。首先，我们概述了数据预处理的步骤，以及如何从原始数据中获得可分析和研究的更易于管理的数据。最后，我们描述了我们对数据的假设。.

4.1数据预处理

通常，分析的很大一部分都处于数据预处理阶段。根据数据，它可以吸收多达50%到80%的总时间来完成分析。在此，我们描述了用于数据的关键处理阶段。

从图1提供的示例数据输出中我们知道，Nessus软件提供的输出属性比我们分析所需的更多。我们的数据预处理最关键的部分是准确地存储和识别独特的漏洞。为了引入一些符号，假设我们将所有漏洞的集合表示为v，然后对于每个唯一的漏洞i，我们将关联的漏洞集合表示为v i。然后我们将唯一的漏洞（v i）定义为四个属性集，即：1.）CVE代码；2.）主机名；3.）端口号；和4.）协议。这四个属性是专门选择的，因为一个漏洞的唯一性不能用少于这四个组合属性的任何东西来确定。例如，并非Nessus输出中的所有漏洞都具有CVE代码。因此，仅仅通过CVE代码识别一个独特的漏洞是不够的。此外，每个主机名都可以显示在具有许多不同端口号和/或协议的扫描中。因此，识别唯一漏洞（vi）的唯一方法是根据CVE代码、主机名、端口号和协议的组合键对数据进行解析。结果值是所有可能的唯一漏洞的集合。然后，我们将这组独特的漏洞进行比较，从一次扫描到另一次扫描，以将其状态定义为新的、现有的、消失的或重新出现的。最后的输出是一组文件，用上述标签表示从一次扫描到下一次扫描的差异。在我们的研究中，我们需要速率，因此进一步将数据转换为计数，然后将总计数除以扫描次数。我们在算法1中总结了数据解析器的源代码，它输出新的、消失的或重新出现的唯一漏洞集。.

4.2假设和关键关系

在本节中，我们将概述数据中的初步假设和关键关系。每个唯一的漏洞都由CVE代码、主机、协议和端口的唯一密钥定义。正如从算法1中可以理解的那样，对数据进行分析的方式使漏洞定义为“新”、“已存在”或“消失”。如下文所述，有一个非常关键的与删除的漏洞相比，消失的漏洞有明显的区别。

从扫描到扫描，漏洞存在于四种可能状态之一，即：

1.新：如果漏洞不是新的，那么从一次扫描到下一次扫描都会被认为是新的。在以前的扫描中观察到。

2.消失：漏洞可以消失，原因有很多，包括服务器为了维护和维护，或甚至在服务器主机命名更改的情况下重新组织网络，已解除授权或处于脱机状态。

3.重新出现：如果漏洞的唯一密钥在扫描中至少出现一次新的漏洞，那么，如果该唯一密钥在任何时间点上连续消失扫描，然后重新出现。

4.已删除：漏洞修补后可以视为已删除。我们收集了一套通过将漏洞消失率与漏洞重新出现率（如删除率）之差来计算已删除漏洞的总体速率，消失率-再现率定义脆弱性比率之间的这种关系删除、消失和再现需要一些简化和假设。我们相信这些简化是建模数据的最佳起点。作为我们模型成熟，我们可以探索合并其他不确定性层的方法，在第6节中进一步说明。

4.3单变量分析

在我们的单变量分析中，我们的目标是通过分布模型描述到达和删除过程。我们评估和比较模型的方法如下:

1. 经验密度和理论密度：我们根据我们认为最适合的理论分布绘制经验数据的直方图。这通常是获得数据直觉的良好起点。Q-Q图（下面描述）提供了对分布是否适合的更深入的了解。
2. 四分位数图（Q-Q图）：Q-Q图比较柱状图的形状和理论密度。如果比较的两个分布相似，则q-q图中的点将近似位于y=x线上。如果分布是线性相关的，则q-q图中的点将近似位于一条线上，但不一定位于y=x线上。
3. 经验和理论CDF：我们比较了经验和理论数据的CDF。我们的目标是寻找一个整体的良好拟合，并尽量减少经验分布和理论分布之间的距离。
4. 概率概率图（P-P图）：P-P图将变量的经验累积分布函数（ecdf）与指定的理论累积分布函数进行比较。这个ECDF定义为从数据中得出的经验累积分布函数。与Q-Q图和概率图一样，P-P图也可以用来确定理论分布对数据分布的建模程度。如果理论CDF在所有方面（包括位置和比例）都合理地模拟了ECDF，则P-P图上的点模式通过原点呈线性，并且具有单位坡度.

4.4二元分析

最终，我们的目标是发现脆弱性出生和死亡过程之间的关系，即到达率和删除率之间的关系，并探索如何将其推广到“野外”的其他脆弱性数据分析中。

我们分析了双变量散点图的趋势，总结了基础统计数据，并评估了其对未来模型的有用性，以及对分析师和安全工程师的实际应用。

5.结果

在本节中，我们首先概述了所有四个数据子集（即新的、删除的、重新出现的和消失的）的一些基本统计特征和分析。然后，我们深入分析了新漏洞的出生率集和删除漏洞的死亡率集。.

最后，我们评估了二元散射图的生灭过程之间的相关性。

5.1汇总统计分析

回顾如图1所示的最终数据集，我们提供了一个漏洞到达、删除和重新出现的散点图。我们发现87%的漏洞由于不同的原因消失，包括删除、服务器脱机或退役或扫描策略更改。我们还发现，73%的漏洞通过修补被删除，14%的漏洞在同一台计算机上的稍后时间点重新出现。

25

Scatterplot of Vulnerability Arrival, Deletion and Reappearance Rate

oe

'

o

p

f

N

t

8

.i.'? if!

e

P N

j ts

.=

,r V%

MS

.'

.Si-

\_:'{!.\_

#':T,S '

'

-

4

-.

"''?

Tt r

0

500

1000

1500

Machine Number

2000

2500

3000

o Arrival

Deletion

Reappearance

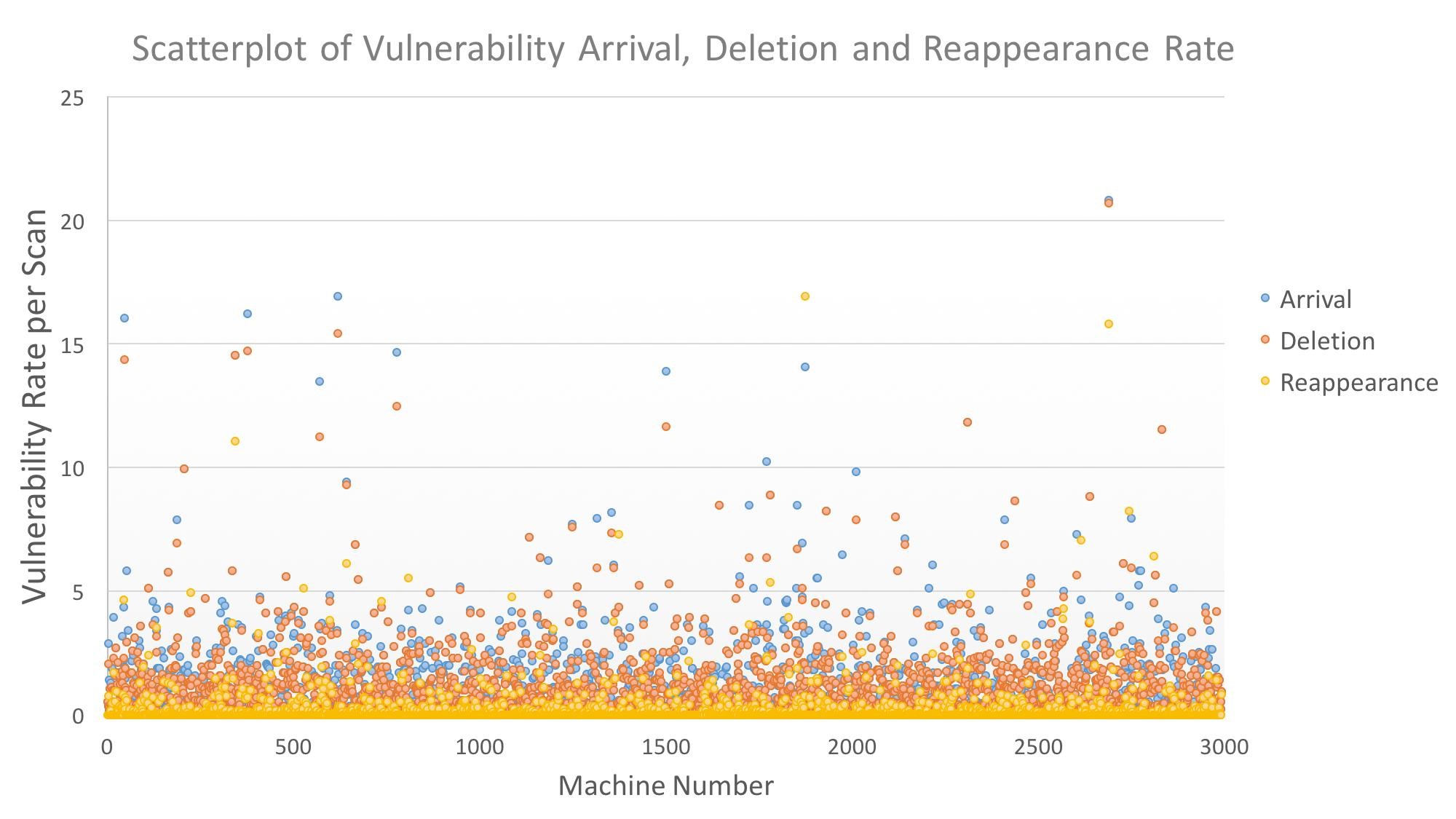


图2：漏洞到达、删除和再现率概述。

在下一节中，我们将对单变量分布和特征进行更深入的检查。

5.2单变量分布和特征

在本节中，我们将概述漏洞到达和删除率分布的近似值，以及统计特征。我们提供了关于漏洞管理的分布适合意味着什么的实际解释。

5.2.1漏洞到达

我们研究了近3000台机器在两年内从17份扫描数据报告中的漏洞到达率。我们将到达率的单变量分布表示为a，这样对于每个主机的单个速率i，我们将每个主机的相关到达率表示为。

我们发现每台主机在至少一个扫描报告上至少有一个漏洞计数；因此，每台机器的最终漏洞到达率大于零。这意味着没有一个宿主能免于妥协的风险。我们还发现，脆弱性到达数据显示了极端右歪斜的特征。这种偏态分布表明低出生率的高计数和极高出生率的低计数。.

表2：漏洞到达率分布的汇总统计数据

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mean | Variance | Standard Deviation | Count |  |
| 1.28 | 2.25 | 1.50 | 2991 |  |
| Minimum | 1st Quartile | Median | 3rd Quartile | Maximum |
| 0.06 | 0.35 | 0.88 | 1.71 | 20.82 |

有趣的是，超过一半的机器（54.90%）的脆弱性出生率小于或等于1，几乎所有机器（97.89%）都小于或等于5。最大值为20.82，这告诉我们刚好超过2%的数据（具体来说，2.11%）说明了（5，20.82）的重尾值范围。进一步的统计调查总结见表2。

我们发现脆弱性到达率满足对数正态分布。图3显示了四个有助于评估分布适合性的图。通过比较脆弱性到达的经验密度和理论密度，验证了对数正态拟合的可靠性。为了更深入的了解，我们发现Q-Q图中的数据非常接近直线，除了几个最右边的圆。当我们用理论CDF来观察经验CDF时，我们看到了一个整体的很好的拟合。最后，经验数据点与P-P图中的直线吻合较好。数据的前半部分有一条温和的曲线，表明经验概率和理论概率之间存在轻微的差异，但总的来说，这是一个很好的拟合。

当我们从对数正态模型的其他领域观察到实例时，对数正态分布是直观的，例如流行病学研究中的病毒生长率、13和机械结构中的疲劳应力失效。14对数正态模型通常最能代表自然现象，其值均大于零。

5.2.1漏洞删除

使用与第5.2.1节中使用的方法类似的方法，我们研究了近3000台机器在两年内从17份扫描数据报告中的漏洞删除率。我们将删除的单变量分布表示为d，这样对我们发现，漏洞删除率集与到达率集的不同之处在于，仅超过22%的机器在两年内完全没有删除漏洞。这表明几乎三分之一的机器都有漏洞，这些漏洞无法以任何方式得到解决。

í?

Ó

u\_

o

U

ó -

o

o

Histogram and theoretical densities

o

co

ó

ó

ó

o

0

5

10

15

Vulnerability Arrivals

20

Empirical and theoretical CDFs

lognormal

i

i

i

i

0

o

ó

ó

o

Q-Q plot

0

5

10

15

20

25

Theoretical quantiles

P -P plot

5

10

15

20

0.0

0.2

0.4

0.6

0.8

1.0

Vulnerability Arrivals

Theoretical probabilities

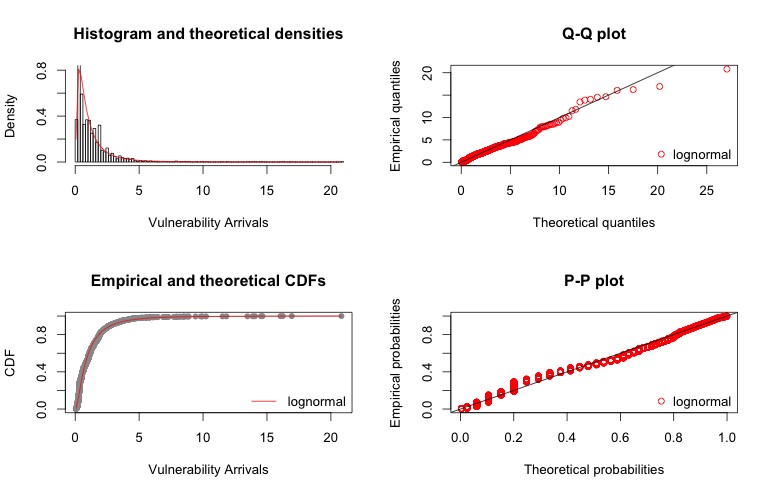


图3：日志正态分布为漏洞到达过程提供了一个可能的模型。

在删除脆弱性率的背景下，造成这种偏差的一个原因可能是所谓的“启动效应”，例如我们在可靠性分析中可能发现的。由于许多不同的可能原因，漏洞修复过程开始时可能会出现大量的“失败”。

毫不奇怪，超过半数的机器（67.20%）的脆弱性死亡率小于或等于1，几乎所有机器（98.36%）都小于或等于5。最大值为20.71时，这告诉我们，小于2%的数据（特别是1.64%）占（5，20.71）的重尾值范围。表3总结了进一步的统计探索。

表3：已删除漏洞率分布的汇总统计数据

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mean | Variance | Standard Deviation | Count |  |  |
| 0.94 | 1.91 | 1.38 | 2991 |  |  |
| Minimum | 1st Quartile | Median | 3rd Quartile |  | Maximum |
| 0.00 | 0.12 | 0.53 | 1.24 |  | 20.71 |

我们还发现，删除的漏洞集可以用指数分布来建模。图4显示了四个有助于评估最佳分布匹配的图。我们发现经验密度和理论密度几乎完全吻合。Q-Q图显示，线的左上角略微向上弯曲，表明底层流程的活动可能比我们目前所能捕获的要多一点。P-P图在左下区域也显示出一些细微的偏差；但是，总体上我们发现指数分布的概率值与经验分布的概率值非常相似。

Histogram and theoretical densities

0

5

I

r

10

15

Vulnerability Deletions

Empirical and theoretical CDFs

20

exponential

0

5

10

15

20

o

Q -Q plot

0

2

4

Theoretical quantiles

P -P plot

6

8

0.0

0.2

0.4

0.6

data

Theoretical probabilities

0.8

1.0

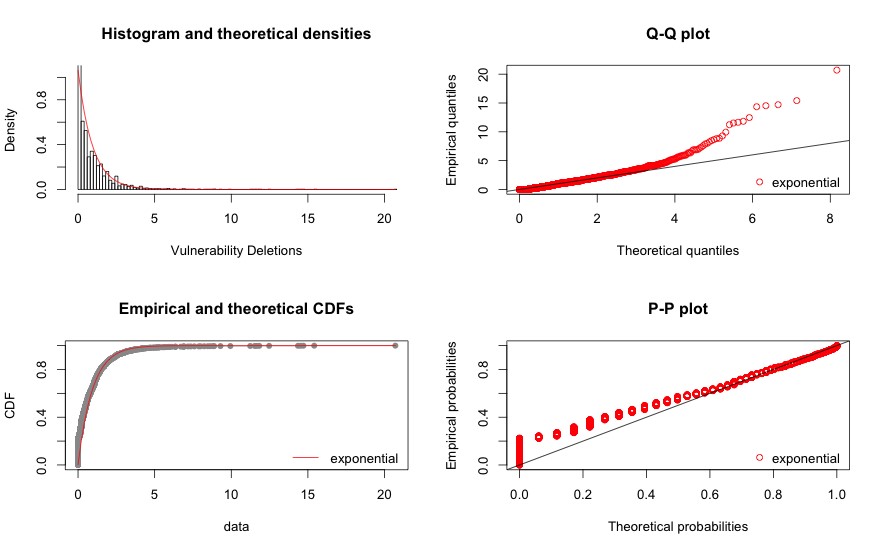


图4：指数分布为漏洞删除过程提供了一个可能的模型

我们怀疑指数分布最适合的原因主要是因为数据中有大量的零。正如我们在第6节中概述的，未来的工作是探索漏洞随着时间的推移会持续多长时间，以及它们的到达率和删除率在扫描之间的变化情况。根据结果，脆弱性生命线可以作为可靠性理论问题来研究。

5.3二元关系

在本节中，我们旨在通过二元散点图和相关统计来说明这种关系。我们知道，漏洞删除在一定程度上依赖于漏洞

†

关于这一问题的进一步评论可以在“未来工作”（第6节）中找到。

到达者在建模这种依赖性时，术语的自然选择是出生率（针对脆弱性到达）和死亡率（针对脆弱性删除）。请注意，在整个分析的这一部分中，我们可以互换使用这种语言。

25

20

15

10

5

-5

0

5

10

15

20

25

Vulnerability Birth Rate per Month, per Host

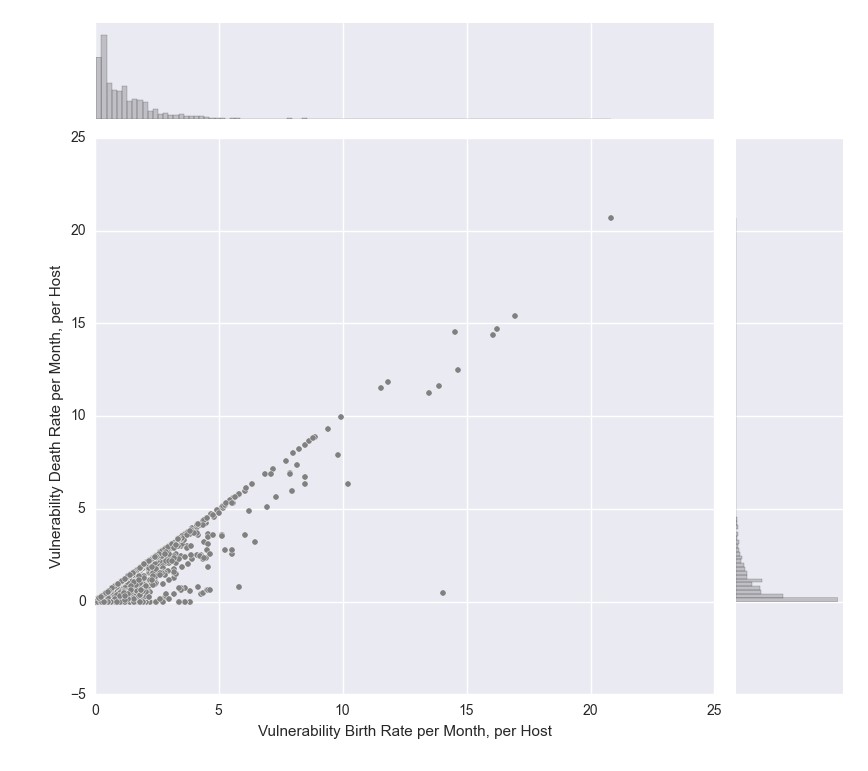


图5：脆弱性出生和死亡率的双变量散点图

表4：出生和死亡率双变量关系的斯皮尔曼相关统

|  |
| --- |
| Spearman Correlation |
| 0.76 |

‡

在图5中，我们看到脆弱性出生率和死亡率之间存在着强烈的正相关关系。表4显示了相关性的进一步证据，其中斯皮尔曼相关性统计为正，与一个相当接近。我们在二元图的x轴和y轴的外边缘包括边缘（单变量）分布，以帮助进一步分析。

我们发现，最好的情况是脆弱性死亡率等于脆弱性出生率。脆弱出生率和死亡率的交点被认为是平衡点。请注意，有些数据点偏离了x=y线。这些数据表明，易受攻击的机器的出生率远远高于它们的死亡率。

这个双变量图的用处在于，它不仅为模型和联合概率分布提供了基础，而且还可以立即向分析人员和安全工程师反馈其脆弱性管理的当前状态。X-Y线上的数据集群越多，我们就知道24个月内修补机器的速度越快。X-Y线下的数据点集群越多，修补漏洞的攻击性就越小，需要维护的漏洞就越大。总之，双变量散点图有助于深入了解漏洞管理的历史。他们还可以根据过去的网络行为对未来的脆弱性指标进行预测分析。

5.4内外部有效性挑战

在本节中，我们回顾了一些已确定的对内部和外部有效性的挑战。认识到并减轻这些挑战对于生成和处理产生可靠结果和彻底科学分析的数据至关重要。正因为如此，我们认识到在我们的研究中，存在着一些对内部和外部有效性的挑战。示例包括但不限于Nessus软件中的误报率或误报率。第5.4.1和5.4.2节全面概述了这些挑战。

5.4.1网络漏洞

内部有效性指的是一个实验进行得如何。支持良好内部有效性的因素具有尽可能少的混淆变量。随着同时作用的自变量数量的增加，对内部有效性的挑战也随之增加。

我们研究的内部有效性挑战包括:

**删除率假设** 我们将删除率定义为漏洞消失率集和漏洞重新出现率集之间的差异。在消失和再现脆弱性之间的关系中可能存在其他混淆变量。尽管如此，我们也不希望过度拟合数据，并且相信我们在计算漏洞删除率集时的准确性相当高。

**由于各种原因**，服务器可以暂时脱机或退役。可能是系统必须升级，或者访问策略更改。如果在数据库中发现“错误”，可能需要实施临时保护措施。在这种情况下，服务器将离线以防止进一步的损坏，直到稍后可以对其进行调查。

**扫描策略更改**企业级网络经历定期更改，例如永久添加和删除服务器。发生这种情况时，需要添加主机名或从漏洞扫描列表中删除主机名。在两年的漏洞数据中，有三次扫描策略更改。

**数据缺口** 24个月中有6个月的缺少扫描结果。我们将数据的结果与缺失月份的各种假设水平进行了比较，发现没有最佳策略来为这个特定的数据集弥补缺失月份。为了避免在数据中引入伪影或无意中扭曲结果，我们决定将数据视为“scan to扫描”，而不是“逐月扫描”（即，我们将丢失的月份单独处理，并将数据视为一组跨越24个月的17次扫描）。

**软件缺陷**，Nessus扫描仪软件本身存在的软件缺陷问题可能导致数据不完整。有一个研究机构致力于解决漏洞软件的误报和误报。我们将好奇的读者引向参考文献3、15和16。

**5.4.2**扩展漏洞

外部效度是指科学结果在类似情况下的概括程度。我们知道，科学研究对其他研究的概括越少，对外部有效性的挑战就越大。

我们研究的外部有效性挑战包括:

**数据**也许对外部有效性最大的挑战是数据。我们正在处理来自一个来源的操作数据。从多个来源获得数据是理想的。由于这项工作是初步研究的一部分，我们将在第6节中讨论将分析扩展到更多的数据源。为了尽量减少这一挑战，部分原因在于我们做出了第4.2节中的假设和概括，以避免数据过度拟合。

6.未来工作

本文主要研究了一个分析大型脆弱性数据集的过程，并探索了一个操作环境下到达和删除过程的统计特征。我们的研究是独一无二的，因为它是第一个从“野外”获取脆弱性数据的来源，随之而来的是大量的安全研究问题，这些问题可以进一步帮助脆弱性缓解和管理过程。具体来说，我们提出了以下研究问题：我们可以量化操作脆弱性生命线吗？根据组织当前的漏洞修补过程，其预计净成本是多少？我们如何衡量或建模风险，以便更有效地进行脆弱性分类？

为了解决这些问题，我们建议将以下工作作为我们结果的一个结果：

* 来源一套更大的操作性IT漏洞扫描数据，以更好地估计当前的漏洞修补状态；
* 通过测量脆弱性持续的时间长度以及时间间隔之间出生-死亡过程的变化，将属性时间（t）纳入我们的模型中；
* 将随机变量时间（t）、脆弱性到达（a）和脆弱性删除（d）纳入改进脆弱性分类的风险模型中；
* 根据如何实时管理漏洞，估计组织的净预期成本。

7.结论

综上所述，本文概述了研究操作系统漏洞的必要性。它提供了一种数据解析算法，用于大型和难以控制的脆弱性数据集中的争论。然后，我们提供了一个深入的单变量和双变量分析，提供了对漏洞到达和删除过程之间关系的洞察。我们用来描述单变量的分布可以用作未来研究的先验分布，例如关于漏洞管理和缓解的贝叶斯预测。未来的工作包括：1）归纳和自动化我们的漏洞分析算法，用于任何环境，从学术到操作；2）量化随机变量时间，（t），通过测量漏洞在时间上的持续性以及时间间隔之间的变化；3）设计风险CA基于三个随机变量到达（A）、删除（D）和时间（T）的LCULU。我们的研究结果自然也有助于为诸如网络保险等领域的大量风险评估和精算研究提供信息。

致谢

这项工作得到了Aro Muri Grant W911NF-13-1-0421的支持。作者感谢国防部提供学生财政支持的智能奖学金，感谢本·普里斯特提供反馈，感谢达特茅斯计算服务提供漏洞数据。

引用

1. Allodi, L. and Massacci, F., “Comparing vulnerability severity and exploits using case-control studies,” *ACM Transactions on Information and System Security (TISSEC)* **17**(1), 1 (2014).
2. Joh, H. and Malaiya, Y. K., “A framework for software security risk evaluation using the vulnerability lifecycle and cvss metrics,” in [*Proc. International Workshop on Risk and Trust in Extended Enterprises*], 430–434 (2010).
3. Holm, H., Ekstedt, M., and Andersson, D., “Empirical analysis of system-level vulnerability metrics through actual attacks,” *Dependable and Secure Computing, IEEE Transactions on* **9**(6), 825–837 (2012).
4. Clark, K., Dawkins, J., and Hale, J., “Security risk metrics: Fusing enterprise objectives and vulnerabilities,” in [*Information Assurance Workshop, 2005. IAW’05. Proceedings from the Sixth Annual IEEE SMC*], 388–393, IEEE (2005).
5. Georg, G., Houmb, S. H., and Ray, I., “Aspect-oriented risk driven development of secure applications,” in [*Data and Applications Security XX*], 282–296, Springer (2006).
6. Carin, L., Cybenko, G., and Hughes, J., “Cybersecurity strategies: The queries methodology,” *Computer* **41**(8), 20–26 (2008).
7. Meier, J., Mackman, A., Dunner, M., Vasireddy, S., Escamilla, R., and Murukan, A., [*Improving web application security: threats and countermeasures*], Microsoft Redmond, WA (2003).
8. Wynn, J., Whitmore, J., Upton, G., Spriggs, L., McKinnon, D., McInnes, R., Graubart, R., and Clausen, L., “Threat assessment and remediation process,” Tech. Rep. Project No. W15P7T-10-C-F600, MITRE (October, 2011).
9. Buckshaw, D. L., Parnell, G. S., Unkenholz, W. L., Parks, D. L., Wallner, J. M., and Saydjari, O. S., “Mission oriented risk and design analysis of critical information systems,” *Military Operations Research* **10**(2), 19–38 (2005).
10. Artz, M. L., *Netspa: A network security planning architecture*, PhD thesis, Massachusetts Institute of Technology (2002).
11. Tenable Network Security, Nessus Vulnerability Scanner https://www.tenable.com/products/nessus-vulnerability-scanner. (Accessed: 18 March 2016).
12. “New Nessus Feature Added: CSV Export.” Tenable Network Security, https://www.tenable.com/blog/new-nessus-feature-added-csv-export. (Accessed: 18 March 2016).
13. Buckland, S. T., Burnham, K. P., and Augustin, N. H., “Model selection: an integral part of inference,” *Biometrics* , 603–618 (1997).
14. Cruse, T. A., [*Reliability-based mechanical design*], vol. 108, CRC Press (1997).
15. Holm, H., Sommestad, T., Almroth, J., and Persson, M., “A quantitative evaluation of vulnerability scanning,” *Information Management & Computer Security* **19**(4), 231–247 (2011).
16. Doup´e, A., Cavedon, L., Kruegel, C., and Vigna, G., “Enemy of the state: A state-aware black-box web vulnerability scanner.,” in [*USENIX Security Symposium*], 523–538 (2012).