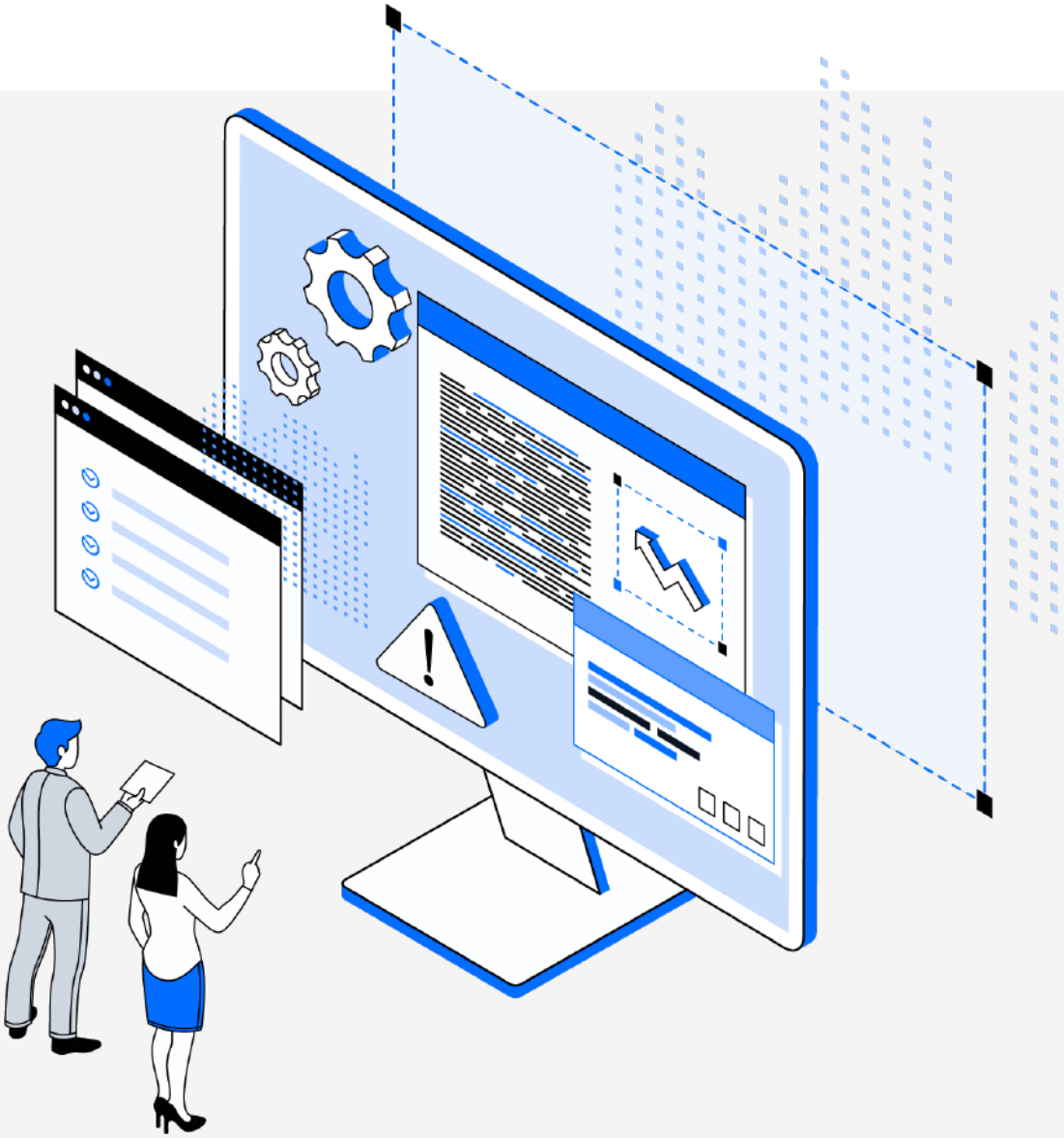
万径安全
MegaVector

渗透测试
质量提升方案



四 维 创 智 （ 北 京 ） 科 技 发 展 有 限 公 司

目录

国内信息安全环境概述

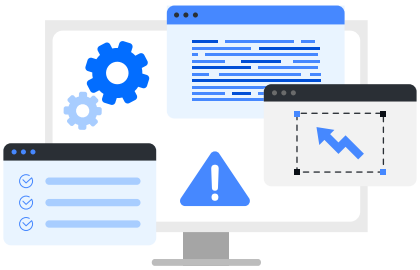
随着互联网的普及，中国在信息技术领域取得了显著进展，但这也带来了新的安全挑战。网络环境面临着大量的攻击和威胁，截至2022年1月，中国已成为全球最活跃的网络攻击目标之一。

为了保障网络安全，规范网络行为，强化网络基础设施，自2017年6月1日起《中华人民共和国网络安全法》正式施行，该法规规定了网络基础设施运营者的责任，包括采取必要的技术措施保护网络安全。政府在信息安全方面的监管一直采取积极的态度，2021年11月1日《个人信息保护法》正式生效，2021年9月1日《数据安全法》正式实施，这三个法规共同构成了中国在网络安全和信息安全领域的法律体系。它们的实施旨在维护国家网络空间安全、保护个人信息隐私，以及规范数据的收集和处理行为。

渗透测试简介

渗透测试，也被称为渗透性评估或者攻击模拟，是完全模拟黑客可能使用的攻击技术和漏洞发现技术，可以对目标系统的安全做深入的探测，发现系统最脆弱的环节。渗透测试能够直观地让管理人员知道自己网络所面临的问题，从而帮助组织提前采取防范措施，避免或者减轻可能发生的损失。

渗透测试是一种非常专业的安全服务。



2.1 渗透测试的重要性

在当今的数字化世界中，网络安全已经成为每个组织、企业乃至个人都无法忽视的重要问题。渗透测试，特别是业务系统上线前的渗透测试，作为网络安全领域的一种主要测试手段，其意义和重要性不言而喻。

渗透测试主要利用网络安全扫描器、专用安全测试工具和富有经验的安全工程师的人工经验对网络中的核心服务器及重要的网络设备，包括服务器、网络设备、防火墙等进行非破坏性质的模拟黑客攻击，目的是侵入系统并获取机密信息并将入侵的过程和细节产生报告给用户。

渗透测试和工具扫描可以很好的互相补充。工具扫描具有很好的效率和速度，但是存在一定的误报率和漏报率，并且不能发现高层次、复杂、并且相互关联的安全问题；渗透测试需要投入的人力资源较大、对测试者的专业技能要求很高（渗透测试报告的价值直接依赖于测试者的专业机能），但是非常准确，可以发现逻辑性更强、更深层次的弱点。

2.2 渗透测试的作用与收益

提高系统安全性
渗透测试能够发现并修复系统中的安全漏洞，从而显著提高系统的安全性， 防止潜在的攻击。
满足合规要求
许多行业和地区的法规要求定期进行渗透测试，以证明组织对信息安全的重视和投入。
防止经济损失
通过渗透测试发现和修复漏洞，可以避免因安全事故导致的巨大经济损失。
保护企业声誉
安全事故往往会对企业的声誉造成严重打击。通过定期的渗透测试，可以有效防止安全事故的发生，保护企业的声誉和客户的信任。
提升安全意识
渗透测试不仅能发现技术漏洞，也能通过模拟攻击的过程，提升组织和员工的安全意识，帮助他们了解和防范潜在的威胁。

2.3 渗透测试面临的挑战和问题

尽管渗透测试在信息安全领域的重要性无可置疑，但在实际操作中，渗透测试仍然面临着许多挑战和问题。

测试工具自身安全的不可控
渗透测试工具往往来自于开源社区或者第三方开发，这些工具的安全性往往无法得到保证。如果测试工具自身存在安全漏洞或者被植入恶意代码，那么在渗透测试过程中可能会给被测试系统带来额外的安全风险。
测试工具版本的不统一
由于各种原因，不同的渗透测试人员可能会使用不同版本的测试工具，这可能会导致测试结果的不一致，或者出现因工具版本过旧而漏报某些新的安全漏洞。
测试过程不可控
在传统的渗透测试中，测试过程往往不透明，无法确保测试人员真的按照预定的测试计划和标准进行全面的测试，这可能会导致一些潜在的安全漏洞被忽视。
测试工具无法原生协同
大多数渗透测试工具是独立开发的，无法实现原生的协同，这给多人协同测试和工具间的数据共享带来了困难，降低了测试的效率。
测试过程数据溯源问题
在渗透测试过程中，可能会产生大量的数据，包括测试结果、测试过程记录、漏洞报告等。如果没有有效的数据管理和溯源机制，那么在后期的分析和复查中，可能会遇到很大的困难。

这些问题不仅影响了渗透测试的效果和效率，也可能给被测试系统带来额外的安全风险。因此，解决这些问题，提升渗透测试的质量和效率，是当前信息安全领域亟待解决的重要任务。

问题描述

3.1 测试工具自身安全的不可控

渗透测试工具是渗透测试过程中不可或缺的部分，它帮助测试人员自动化或半自动化地发现并利用系统的安全漏洞。然而，测试工具自身的安全性问题，却是一个经常被忽视的风险点。

测试工具自身的安全性问题主要体现在两个方面：

① 工具自身的安全漏洞

许多渗透测试工具，尤其是使用量较大的开源工具或者商业化闭源工具，可能自身存在安全漏洞。这些漏洞可能被恶意攻击者利用，进而影响到渗透测试的安全性。攻击者可能通过工具的漏洞控制测试工具，进而对被测试系统进行恶意攻击。

Burp Suite v2.0 内置的 Chromium 版本为64.0.3282.24，该低版本 Chromium 受到多个历史漏洞影响，用户在通过 Render 功能渲染页面时触发 v8 漏洞成功执行 shellcode，从而进一步获得 PC 权限。

AWVS V10.5 也曾出现一个本地提权漏洞，漏洞是出现在 AWVS 10 的一个任务调度的 API 上。在 AWVS 10.x 被安装后，系统会默认安装一个叫做 “AcuWVSSchedulerv10” 的自启动服务，这个服务是跑在 system 权限下的。它会监听本地的8183端口，用户可以通过它直接调用接口来给 AWVS 添加新的扫描任务。在添加任务时，参数里有一项为 reporttemplate，它的作用是选择扫描结束生成报告时所用的模板，研究发现这个参数会被带入 AWVS 的命令行执行。 由于系统没有对用户的输入做检查，导致我们可以通过 reporttemplate 带入任意的参数，形成了命令注入。

在2022年攻防演练期间，Cobalt Strike 也被爆出存在远程代码执行漏洞 (CVE - 2022 - 39197) 。该漏洞存在于 Cobalt Strike 的 Beacon 软件中，可能允许攻击者在 Beacon 配置中设置格式错误的用户名，触发 XSS，从而导致在 CS 服务端上造成远程代码执行。

① 工具被植入恶意代码

在某些情况下，渗透测试工具可能被植入恶意代码。这可能是由于工具开发者恶意植入，也可能是工具在分发过程中被篡改。无论是哪种情况，恶意代码都可能在渗透测试过程中对被测试系统造成损害。

2015年，一个被修改版本的 Apple 的 Xcode 开发工具在中国的网络上流传开来。这个被修改的版本被称为 XcodeGhost，它会在开发者使用它来构建应用程序时，把恶意代码插入到应用程序中。当用户下载并使用了这些被投毒的应用程序后，恶意代码就会在用户的设备上运行。

供应链投毒事件不仅发生在开发人员常用的工具领域，在渗透测试工具领域也时有发生，如大家最长使用的 BurpSuite、CobaltStrike 工具均为商业化闭源工具，工作人员往往通过破解器破解后使用，而在破解器中存在极大的投毒风险。由于渗透测试人员工作的特殊性，往往可以解除更高权限的网络环境，其电脑权限价值往往高于普通的程序员。

为了解决以上问题，万径安全建议：

使用权威来源的工具

尽可能从权威和可信的来源获取渗透测试工具，避免使用来源不明的工具。

定期更新工具

定期更新渗透测试工具，以获取最新的安全补丁和功能更新。

对工具进行安全审计

在使用新的渗透测试工具之前，应进行安全审计，检查是否存在安全漏洞或被植入恶意代码。

使用沙箱运行工具

在可能的情况下，可以考虑使用沙箱或其他隔离技术运行渗透测试工具，以防止可能的安全风险影响到被测试系统。

通过这些措施，可以有效降低渗透测试工具自身的安全风险，从而提升渗透测试的安全性和效果。

3.2 测试工具版本的不统一

渗透测试过程中，测试工具的版本统一性是一个非常重要且容易被忽视的问题。由于个人习惯、知识水平、获取资源的便利性等各种原因，不同渗透测试人员可能会使用不同版本的测试工具。

可能导致：

测试结果的不一致性

不同版本的测试工具可能会有不同的测试结果。例如，新版的工具可能会包含对最新漏洞的检测，而旧版的工具可能无法检测到这些漏洞。这种情况可能会导致测试结果不一致，影响到渗透测试的准确性和完整性。

工具维护的复杂性

如果团队中的每个人都使用不同版本的工具，那么工具的维护工作将会变得非常复杂。例如，当发现一个工具的安全问题时，可能需要在多个版本中都进行修复。

为了解决以上问题，万径安全建议：

统一工具版本

团队应该统一使用相同版本的渗透测试工具。这可以通过建立内部的工具库，提供统一的工具下载和更新服务来实现。

定期更新工具

团队应该定期更新渗透测试工具，以获取最新的安全补丁和功能更新。同时，应该确保所有人都使用了最新的工具版本。

提供工具使用培训

对于新的工具版本，团队应该提供相应的使用培训，以确保所有人都能正确地使用工具。

通过这些措施，可以有效地解决测试工具版本不统一的问题，从而提升渗透测试的准确性和效率。

3.3 测试过程不可控

渗透测试过程的可控性是提升测试质量的关键因素之一。然而，在实际操作中，由于各种原因，包括但不限于测试人员的技术水平、测试环境的复杂性、测试工具的限制等，测试过程的可控性往往难以保证。

可能导致：

测试结果的不可预测性

如果测试过程不可控，那么测试结果也将变得不可预测。这不仅会影响到渗透测试的准确性，也可能会对被测试系统造成不必要的风险。

测试效率的低下

不可控的测试过程往往会导致测试效率的低下。例如，测试人员可能需要花费大量时间处理意外的问题，而无法专注于实际的测试工作。

为了解决以上问题，万径安全建议：

建立标准化的测试流程

团队应建立标准化的渗透测试流程，包括测试前的准备工作、测试过程的执行步骤、测试后的报告编写等。这可以帮助测试人员更好地控制测试过程，提高测试效率。

提供专业的培训

团队应该为测试人员提供专业的渗透测试培训，提升技术水平和问题处理能力。这不仅可以提高测试过程的可控性，也可以提高测试的准确性。

使用高质量的测试工具

团队应该选择高质量的渗透测试工具，以减少工具本身可能导致的问题。同时，应该定期更新和维护工具，以确保其稳定性和可靠性。

3.4 测试工具无法原生协同

在渗透测试中，协同工作是非常重要的。然而，很多渗透测试工具都是设计为单一用户、解决单一场景问题使用，而非团队协作。这就导致了测试工具无法原生支持协同工作。

可能面临的问题：

数据共享的困难

如果工具无法支持协同工作，那么测试人员之间的数据共享就会变得困难。 例如，一个人发现的漏洞可能需要手动通知其他人，而不能直接通过工具进行共享。

协作效率低下

如果工具无法支持协同工作，那么团队的协作效率就可能会降低。例如，每个人都需要单独进行相同的测试步骤，而不能共享测试结果。

为了解决以上问题，万径安全建议：

选择支持协同工作的工具

在选择渗透测试工具时，应该优先选择支持协同工作的工具。这样可以直接通过工具进行数据共享和协作，提高团队的工作效率。

建立数据共享机制

即使工具本身不支持协同工作，团队也应该建立数据共享机制。例如，可以使用共享文档或数据库来存储和共享测试结果。

进行协同工作培训

团队应该为测试人员提供协同工作的培训，让他们了解如何有效地进行团队协作。

通过这些措施，可以有效地解决测试工具无法原生协同的问题，从而提升渗透测试的质量和效率。

3.5 测试过程数据溯源问题

在日常工作中，当监管单位或主管领导通过其他渠道发现新的漏洞并下发漏洞整改通知时，测试团队常常发现当时确实对目标进行过安全测试，但是无法还原追溯当时的测试人、测试过程，无法解释当时为何没有测试出对应的漏洞，也就没办法沉淀测试经验实现测试效能的成长。数据溯源可以帮助我们理解测试结果的来源，解释测试过程中出现的问题，以及验证测试结果的准确性。然而，由于各种原因，包括但不限于测试工具的限制、测试环境的复杂性、测试人员的操作失误等，渗透测试过程中的数据溯源往往存在问题。

可能导致：

测试结果的不可验证性

如果无法追溯测试数据的来源，那么测试结果的准确性就无法得到验证。这可能会导致错误的决策和不必要的风险。

问题的难以定位

如果无法追溯测试数据的来源，那么在测试过程中出现的问题就难以定位。这可能会导致问题的解决被延误，影响测试的效率。

为了解决以上问题，万径安全建议：

记录详细的测试日志

团队应该记录详细的测试日志，包括测试过程中的每一步操作、每一个测试结果以及每一个出现的问题。这可以帮助我们追溯测试数据的来源，验证测试结果的准确性，以及定位测试过程中出现的问题。

使用支持数据溯源的工具

在选择渗透测试工具时，应优先选择支持数据溯源的工具。这样可以直接通过工具进行数据溯源，提高数据溯源的效率。

提供数据溯源的培训

团队应该为测试人员提供数据溯源的培训，让他们了解如何有效地进行数据溯源。

通过这些措施，可以有效解决渗透测试过程中的数据溯源问题，从而提升渗透测试的质量和效率。

问题分析

4.1 测试工具自身安全的不可控的风险分析

在渗透测试中，测试工具自身的安全性是一个重要的考虑因素。如果工具自身存在安全漏洞，那么它可能被利用来攻击测试者的系统，或者泄露敏感信息。然而，由于各种原因，包括但不限于工具开发者、工具的更新频率、工具的复杂性等，测试工具自身的安全性往往是不可控的。这可能导致：

系统被攻击的风险

如果工具自身存在安全漏洞，那么它可能被利用来攻击测试者的系统。这可能导致系统数据被泄露，或者系统功能被破坏。

敏感信息被泄露的风险

如果工具自身存在安全漏洞，那么它可能被利用来泄露测试过程中获取的敏感信息。这可能导致企业的商业秘密被泄露，或者客户的隐私被侵犯。

4.2 测试工具版本不统一的影响分析

在渗透测试过程中，使用的工具版本不统一可能会带来一系列的问题。这主要是因为不同版本的工具可能存在功能差异、漏洞修复程度不同、操作方法不一致等因素。这可能会出现一些不可控的影响：

① 测试结果不一致

如果团队中的人员使用的是不同版本的工具，那可能会得到不一致的测试结果。这是因为不同版本的工具可能存在功能差异，导致测试结果的不一致。

① 工具漏洞利用不一致

不同版本的工具可能修复了不同的漏洞，这可能导致一些测试人员能够利用某些漏洞，而其他人则不能。

① 操作方法不一致

不同版本的工具可能有不同的操作方法，这可能导致测试人员需要花费更多的时间和精力来学习和适应。

① 工具维护困难

如果团队中的人员使用的是不同版本的工具，那么工具的维护工作就会变得更加困难。这是因为需要针对不同版本的工具进行不同的维护工作。

4.3 测试过程不可控的风险分析

在渗透测试中，如果测试过程不可控，可能会带来一系列的风险。这种不可控性可能源于多个因素，包括但不限于测试环境的不稳定性、测试工具的不可预测性、测试人员的技术水平差异等。这些问题可能会造成一些风险：

① 测试效果不稳定

如果测试环境不稳定，或者测试工具的行为不可预测，可能会导致测试结果的不稳定。这可能会影响到渗透测试的有效性和准确性。

① 潜在的系统损坏风险

如果测试过程不可控，可能会导致系统受到意外的损坏。例如，一个不稳定的测试工具可能会导致系统崩溃，或者一个不精确的测试可能会误删除重要的系统文件。

① 测试人员技术水平差异导致的风险

如果测试人员的技术水平差异较大，那么他们可能会在测试过程中产生不一致的测试效果。这可能会导致测试结果的不准确，或者测试过程的效率降低。

4.4 测试工具无法原生协同的效率问题分析

在渗透测试中，使用的各种工具如果无法原生协同，可能会严重影响测试的效率。这可能导致：

重复工作

如果工具之间无法共享信息，可能会导致测试人员在不同的工具之间进行重复的数据输入和结果分析，增加了不必要的工作负担。

信息丢失或不一致

在无法进行有效协同的工具之间传递信息可能会导致信息丢失或不一致。这可能会影响到测试结果的准确性和完整性。

工作流程中断

如果一个工具的结果不能直接被另一个工具使用，可能会导致工作流程的中断，增加了测试过程的复杂性。

效率降低

所有这些问题都可能导致渗透测试的效率降低。这不仅会增加测试的时间和成本，还可能影响到测试的质量。

为了解决这些问题，需要寻找可以原生协同的工具，或者开发工具间的接口以实现信息的有效传递。

4.5 测试过程数据无法溯源的监管问题分析

在渗透测试过程中，如果无法跟踪和记录测试数据的来源，可能会引发一系列的监管问题：

无法追踪问题来源

如果测试过程中出现问题，但无法准确追踪到问题的来源，这会导致问题本身得不到解决抑或延期解决，从而影响到测试的效率和质量，更甚者可能导致公司业务不能顺利上线。

无法进行有效的审计

如果无法追踪测试数据的来源，可能会使得审计过程变得很困难。这可能会影响到组织的合规性，甚至可能导致法律问题。

无法进行有效的质量控制

如果无法追踪测试数据的来源，可能会使得质量控制变得困难。这可能会影响到测试的质量，从而影响到测试的效果。

解决方案提出

5.1 重构底层网安能力，打造一体化测试平台

为了解决渗透测试过程中的问题，Yak 语言实现对底层的网络安全能力的重构，并打造了一个一体化的测试平台: Yakit 。

使用者可以在一个视窗内，完成渗透测试全流程的动作执行，不需要安装其他三方依赖，工具原生适配和协同。

工具的渗透测试能力源于 Yak 引擎针对网络安全能力的函数级融合，通过 Yakit 可以实现一体化的测试机制。

统一的测试工具

Yakit 能够统一管理和执行各种渗透测试任务的工具。工具覆盖信息收集、爬虫、交互式 WEB 流量分析、端口扫描、口令破解等，能够满足各种渗透测试的需求，并且能够在工具之间进行有效的协同。

数据追踪和管理

在 Yakit 里有项目管理的功能，在一个项目中产生的所有业务流量均会被记录到单独的数据库中，该数据库可以自动上传至统一的离线 Server 之上，实现数据的追踪和管理，也可以为后续的质量分析提供数据支撑。

自动化和智能化

Yakit 中的所有插件，包括 MITM 插件、端口扫描插件，均可以原生进行自动化执行。用户甚至可以将爬虫和 MITM 插件进行结合，自动化挖掘漏洞。同时， Yakit 中原生集成自研网络安全大模型“万径千机(ChatCS) ”，在测试过程中遇到的所有问题均有专业的 AI 机器人进行高效回复。

安全性和可靠性

Yakit 所有的核心能力均源自于自研的 Yaklang 开发语言，该语言为业内首个致力于网络安全能力融合的开发语言，完全自主可控，所有50万余行代码均自主开发，确保了工具自身的安全性和可靠性。

5.2 通过一体化平台提升测试效率和效果

一体化的渗透测试平台可以大大提升测试的效率和效果。

提升测试效率

通过统一的测试工具和自动化的测试过程，可以大大提升测试的效率。统一的测试工具可以简化测试过程，减少手动操作的需要。自动化的测试过程可以降低人工错误的可能性，提升测试的速度。

提升测试质量

通过数据追踪和管理，可以提升测试的质量。数据追踪可以帮助渗透测试人员准确地找到问题来源，从而更有效地解决问题。数据管理可以帮助人员更好地理解测试过程，从而优化测试策略。

提升测试的智能化水平

通过利用人工智能和机器学习技术，可以提升测试的智能化水平。例如，使用大模型的能力来自动分析测试结果，从而快速发现潜在的安全问题。

提升测试的安全性和可靠性

通过确保测试平台的安全性和可靠性，可以提升测试的信任度。这包括保护测试数据的安全，防止未经授权的访问，以及确保测试过程的稳定性。

通过一体化的渗透测试平台，可以大大提升渗透测试的效率和效果，从而更好地保护网络安全。

5.3 实现隔离内网环境中的多人协同

在渗透测试的过程中，多人协同是非常重要的一环。在隔离的内网环境中，由于网络环境限制，用户之间无法及时有效沟通，阻碍了测试效率提升。

测试数据实时共享

用户可以将不同的 Yakit 客户端连接到统一的 grpc 服务端，实现测试流量和测试过程的实时共享，即时掌握同组同事的测试进度和测试细节。

WebFuzzer 数据共享

测试团队之间可以通过一个简单的分享密令随时共享对某一个漏洞的测试细节数据包。方便在需要帮助时，通过共享测试数据快速寻求帮助。

远程协助功能

在隔绝内网中，通过统一协作服务器的配置，多个用户之间可以随时完成远程协助的功能，类似在内网部署一个可即时分享屏幕的向日葵远程协助工具。

实现工具的统一和协同

所有的工具都能够在隔离的内网环境中正常工作，并且能够进行有效的协同。这包括测试工具、通信工具、数据管理工具等。

5.4 团队共享所有的测试脚本插件

在渗透测试中，测试脚本和插件是非常重要的工具。Yakit 企业版提供了强大的插件功能，用户之间可以共享所有的测试脚本和插件，团队成员可以更好地协作，并提高测试的效率和效果。

建立共享库

在 Yakit 企业版中，建立了一个插件共享库，用于存放所有的测试脚本和插件。支持实时的更新和访问，以便于团队成员随时获取和使用最新的脚本和插件。

实现版本控制

实现脚本和插件的版本控制。这可以帮助团队成员跟踪脚本和插件的修改历史，从而更好地理解和管理这些工具。

建立脚本和插件的标准

Yakit 具备完善的脚本和插件的编写和使用标准。这可以保证团队成员的脚本和插件质量，同时也可以帮助新的团队成员更快地理解和使用这些工具。

实现脚本和插件的评审和测试

在 Yakit 中，具备完善的脚本和插件的评审和测试流程。管理员可以控制每一个插件工具的使用权限，这可以帮助团队成员发现和修复问题，从而提高脚本和插件的效果。

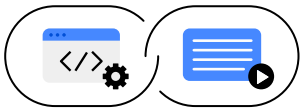
通过实施这些方法，Yakit 可以实现团队共享所有的测试脚本和插件，从而提升渗透测试的效率和效果。

5.5 使用国产自研语言建立贴合企业的安全体系

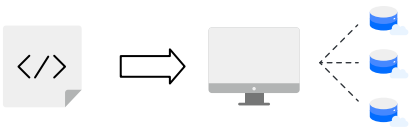
Yakit 的所有能力来自于国产自主开发语言 Yak，这是 CDSL（Cybersecurity Domain Specifi-cLanguage，网络安全领域编程语言）在网络安全行业的首次尝试和落地。

Yak 是一门针对网络安全领域研发的易书写、易分发的高级计算机编程语言。Yak 具备强类型、动态类型的经典类型特征，兼具编译字节码和解释执行的运行特征。

Yak 语言的运行环境只依赖于 YakVM，可以实现“一次编写，处处运行”的特性，只要有 YakVM 部署的环境，都可以快速执行 Yak 语言程序。



实现 Yak 语言在编译时与运行时分离



把 Yak 代码编译成 YakVM 字节码
且字节码可持久化分发到不同平台

Yak 语言起初只作为一个“嵌入式语言”在宿主程序中存在，后在电子科技大学网络空间安全学院学术指导下，由 Yaklang.io 开源团队进行了长达两年的迭代与改造，实现了 YakVM 虚拟机让语言可以脱离“宿主语言”独立运行，并与2023年完全开源。目前支持主流操作系统: MacOS，Linux，Windows。

基于 CDSL 概念构建的网络安全领域编程语言 Yak，具备了 DSL 所有的优势，主要应用于安全能力研发领域，内置库函数封装了现有安全市场上的大多数产品功能，可以让各种各样的安全能力彼此之间“互补、融合、进化”，提高安全从业人员的生产力。

CDSL 在网络安全领域提供的能力具备很多优势：

简洁性

使用 CDSL 构建的安全产品更能实现业务和能力的分离，并且解决方案更加直观。

易用性

非专业的人员也可以使用 CDSL 构建安全产品，而避免安全产品工程化中的信息差。

灵活性

CDSL 一般被设计为单独使用和嵌入式使用均可，用户可以根据自己的需求去编写 DSL 脚本以实现特定的策略和检测规则，这往往更能把用户的思路展示出来，而不必受到冗杂知识的制约。

作为一门专为网络安全研发设计的语言，Yak 除了满足一些基础的语言本身需要具备的特性之外，还具有很多特殊功能。

它可以帮助用户快速构建网络安全应用，例如：

- 中间人劫持库函数
- 复杂端口扫描和服务指纹识别
- 网络安全领域的加解密库

Yak 语言还支持中国商用密码体系：支持 SM2 椭圆曲线公钥密码算法， SM4 分组密码算法，SM3 密码杂凑算法等。

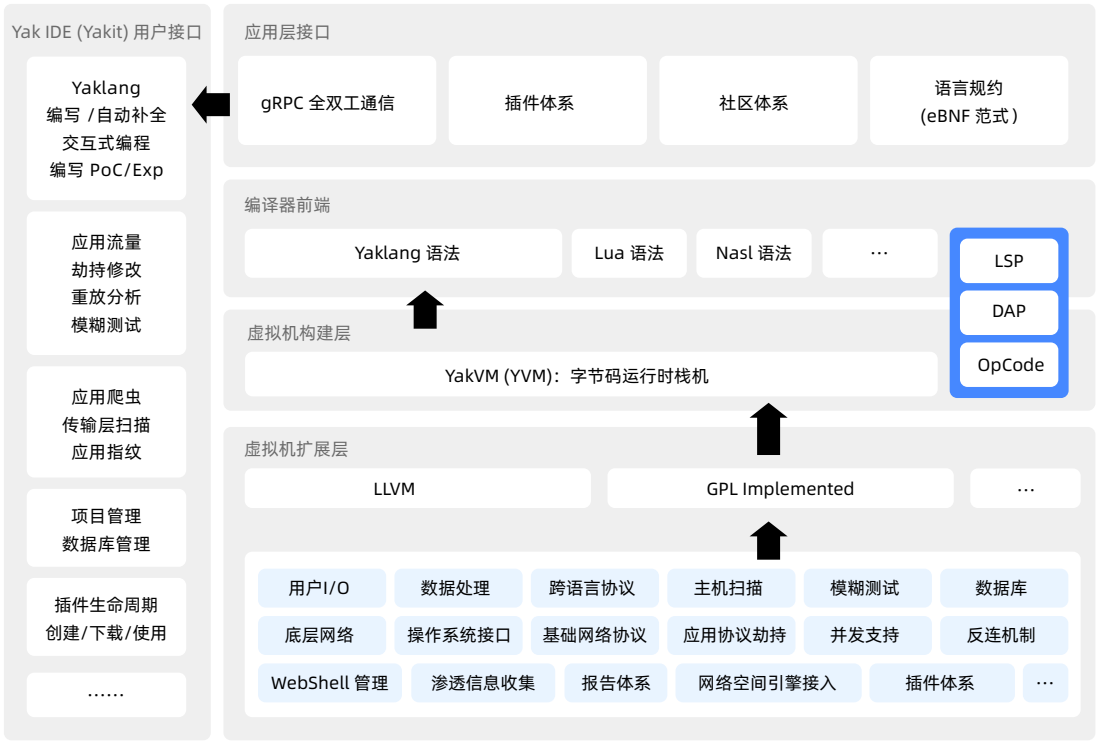
100+安全领域
专用模块

700+可用接口

2000+安全领域
专用库函数

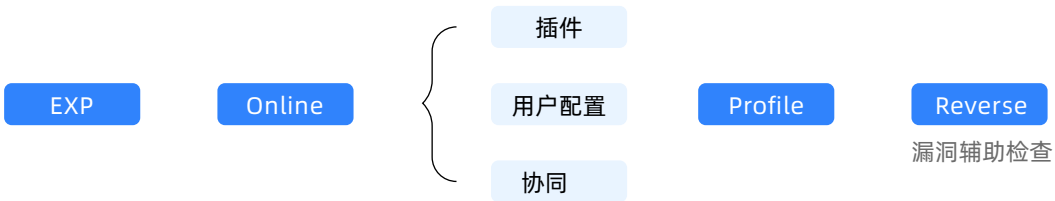
国密算法支持

5.6 Yak 语言的整体架构



解决方案的部署实施

针对企业使用，Yakit 提供了一套完整的解决方案。包括身份认证、管理以及协同的功能，让企业更好地管控数据、管理人员、沉淀知识。同时为了数据的安全性，插件商店及反连平台 (漏洞检测辅助) 可以进行私有化部署，部署在企业自有的服务器上。Yakit 也可根据企业的需要提供 EXP 服务、菜单配置等用户配置服务，以及为企业的实际使用场景提供定制服务。



预期效果

7.1 提升测试工具的安全性

通过在渗透测试环节引入 Yakit 交互式协同测试工具，替代掉原有零散的工具，可确保测试内部不再因工具自身安全引发的重大风险问题。

（1）工具的安全审查

团队本身已完成对 Yakit 的安全测试，也经过了社区用户几万人的持续验证， 确保所有工具的代码质量，排除可能存在的安全风险。此外，团队技术人员也会定期对工具进行安全审计，发现并修复新出现的安全问题。

（2）工具的统一管理

通过 Yakit 工具的统一部署，可以确保所有的测试人员使用同一工具。用户只需学习和熟悉一个工具集，就能完成多项任务，减少在不同工具之间切换的时间和精力，从而提高工作的整体效率。

（3）针对工具的安全培训

万径安全还会不定期对使用这些工具的人员进行安全培训，确保用户熟悉并安全地使用这些工具，以便在发现安全问题时及时报告。

7.2 统一测试工具版本，减少漏报风险

测试工具的版本不一致可能会导致测试结果的不一致，甚至漏报重要的安全问题。通过 Yakit 工具的统一部署和管理，可以确保所有员工使用的都是统一的版本。

（1）工具更新的自动化

可以利用 Yakit 实现工具更新自动化。当工具的新版本发布时，所有测试人员均可以自动获取并部署新版本，确保在任何时候都使用最新的工具进行测试。

（2）工具版本的跟踪和记录

系统会跟踪和记录所有工具的版本信息。这样可以帮助企业在内部系统出现问题时及时排查，在需要时，能够快速回滚到之前的版本。

通过以上措施，可以减少由于版本不一致导致的漏报风险。这不仅可以提高渗透测试的准确性和可靠性，还可以提高员工的工作效率。

7.3 提升测试过程的可控性

企业上层通过 Yakit 工具能实现一系列的管理策略，可以更好地控制和监视测试过程，确保其工作按照预定的流程和标准进行。

（1）测试流程的标准化

通过 Yakit 工具，可以保证测试流程的标准化实施，也可以帮助企业更有效地管理和控制测试过程，确保所有的测试都按照同一套流程进行。

（2）测试过程的监视

利用 Yakit 工具进行实时的测试过程监视。帮助企业及时发现并解决测试过程中可能出现的问题，从而提高测试的效率和准确性。

（3）测试结果的记录和分析

记录和分析所有的测试结果，不仅可以帮助企业了解测试的效果，而且可以帮助企业发现并改进测试过程中存在的问题。

（4）测试人员的培训和指导

对参与测试的人员进行培训和指导，确保他们了解测试流程和标准，以及如何在测试过程中应对可能出现的问题。

7.4 提升测试工具的协同效率

通过使用 Yakit 工具和实施一系列的管理策略，可以更好地协调各种测试工具，提高它们的协同效率。

（1）工具集成的优化

可以实现各种测试工具的深度集成。这不仅可以减少工具之间的兼容冲突，而且可以提高工具的协同效率，如爬虫和 MITM 插件的配合、端口扫描和插件的配合等。

（2）测试数据的协同共享

用户可以接入统一的 Server，实现测试过程中所有数据的共享共通，提升测试效率。

（3） WEB 流量共享和远程协助

通过 Yakit 的 WEB 流量离线共享功能和远程协助功能，可以使得新手学员向高水平测试人员快速分享问题和寻求解决方案。

Yakit 能够确保企业有效地协调使用各种测试工具，这不仅可以大幅度提升测试工具的协同效率，还可以提高测试的准确性和可靠性，提高员工的工作效率。

7.5 提升测试过程数据的溯源能力

通过使用 Yakit 工具和实施一系列的管理策略，可以更好地追踪和管理测试过程数据，提高其溯源能力。

（1）数据记录的完整性

Yakit 可以实现对测试过程中所有数据的完整记录。帮助企业了解测试过程的详细情况，为数据溯源提供强大的支持。

（2）数据记录的一致性

Yakit 可以确保数据记录的一致性和准确性，从而减少由于数据不一致导致的安全问题。

（3） 数据分析的深度

利用数据分析工具对测试过程数据进行深度分析，发现并解决测试过程中可能存在的安全风险问题。

（4） 数据使用的培训和指导

对数据分析人员进行培训和指导，让他们了解如何正确地使用这些数据，以及如何在使用过程中进行溯源。

通过以上的措施，可以显著提升测试过程数据的溯源能力。这可以帮助企业更好地理解和改进测试过程，确保在问题追责中，能够有效地追踪和管理测试过程数据。

万径安全
渗透测试质量提升方案

万径安全
MegaVector

万径安全
MegaVector

让 世 界 更 安 全　　让 安 全 更 简 单

YAK公众号

万径安全公众号

010-5945 6626（北京）
北京市海淀区 上地街道 金隅嘉华大厦 F座804
market@4dogs.cn
http://megavector.cn