软件所在正则表达式拒绝服务攻击漏洞检测与修复技术研究中取得进展

作者：writer 来源：中国科学院

本文原地址：https://www.iikx.com/news/progress/14287.html

近日，中国科学院软件研究所研究员陈海明团队在正则表达式拒绝服务攻击（ReDoS）漏洞检测与修复技术研究中取得重要进展，设计研发出当前最先进的ReDoS漏洞检测工具——ReDoSHunter，提出首个抗ReDoS漏洞正则表达式修复工具——FlashRegex，不仅解决了静态与动态ReDoS检测工具的局限性，实现了正则表达式修复结果无ReDoS漏洞的进展，而且在性能上大幅提升且全面超越了ReDoS漏洞检测及修复工具，为ReDoS漏洞挖掘、利用、修复及防御工作提供了便捷、高效、性能优越的重要工具。

ReDoS漏洞检测工具——ReDoSHunter

正则表达式在计算机科学领域中被广泛使用，但正则表达式拒绝服务攻击（ReDoS）漏洞是一种常见且严重的算法复杂度攻击漏洞，并在近几年呈增长趋势。然而，现有的ReDoS漏洞检测工具存在准确率较低（误报多）或召回率较低（漏报多）的缺陷，产生这一缺陷的根本原因在于，给出全面的、形式化的ReDoS漏洞检测条件这一挑战性问题未得到解决。

针对上述问题，陈海明团队经过长期深入研究，通过对海量易受ReDoS漏洞攻击的正则表达式的分析，创新性地提出了ReDoS漏洞检测条件——ReDoS漏洞模式，并形式化地给出了触发这些模式的必要条件。基于上述工作，进一步提出了动静态结合的ReDoS漏洞检测算法，并设计实现了ReDoS漏洞检测工具——ReDoSHunter。

ReDoSHunter能够高效检测ReDoS漏洞，实现诊断漏洞根本原因、分析漏洞严重程度、追踪漏洞位置并生成触发攻击的字符串等功能。在检测ReDoS漏洞数量方面，ReDoSHunter超越现有最先进工具，在Corpus、RegExLib、Snort三个大型数据集（共计37651个正则表达式）上实现了100%的准确率和召回率。在检测ReDoS漏洞相关的CVEs（Common Vulnerabilities and Exposures通用漏洞披露）方面，现有最先进的检测算法只能检测出60%的ReDoS相关的CVEs，ReDoSHunter能够成功检测出100%的CVEs。由于ReDoSHunter的卓越性能，目前软件所在ReDoS相关的CVEs披露数量排名中位居国际首位。

此外，ReDoSHunter的应用对寻找、纠正广大的开源社区的ReDoS漏洞发挥了重大作用。该工具已应用在Python源码、CKEditor和prismjs等开源项目的ReDoS漏洞检测中。同时，该团队与Snyk建立了长期合作关系，共同致力于高效披露ReDoS漏洞。在已发现的200多个尚未被披露的漏洞中，截至目前已获批了27个CVEs，并收到多个项目的官方致谢。相关研究成果以*ReDoSHunter: a combined static and dynamic approach for regular expression DoS detection*为题，被*USENIX Security*

*2021*会议录用。

抗ReDoS漏洞正则表达式修复工具——FlashRegex

正则表达式以难以掌握著称，其现有的自动化合成与修复工作均忽略了ReDoS漏洞，使其结果可能受到该漏洞的攻击。

针对该问题，陈海明团队提出了首个抗ReDoS漏洞正则表达式的合成与修复算法，其通过去除正则表达式的歧义来生成或修复出无ReDoS漏洞的正则表达式。为加速合成和修复过程，团队使用了确定性自动机和局部约束加强启发式策略，并且设计实现了相应的工具——FlashRegex。

与传统的人工修复相比，采用维护人员的修复方案得到的结果仍常有ReDoS漏洞，而FlashRegex能够高效地生成或修复出无ReDoS漏洞的正则表达式，修复的所有正则表达式中发现的ReDoS漏洞数量为0。该工具已应用到实际开源项目中修复ReDoS漏洞，得到postccs、nltk和Python源码等多个项目维护者及Snyk的认可或致谢。相关研究成果以*FlashRegex: deducing anti-ReDoS regexes from examples*为题，发表在*ASE 2020*会议上。

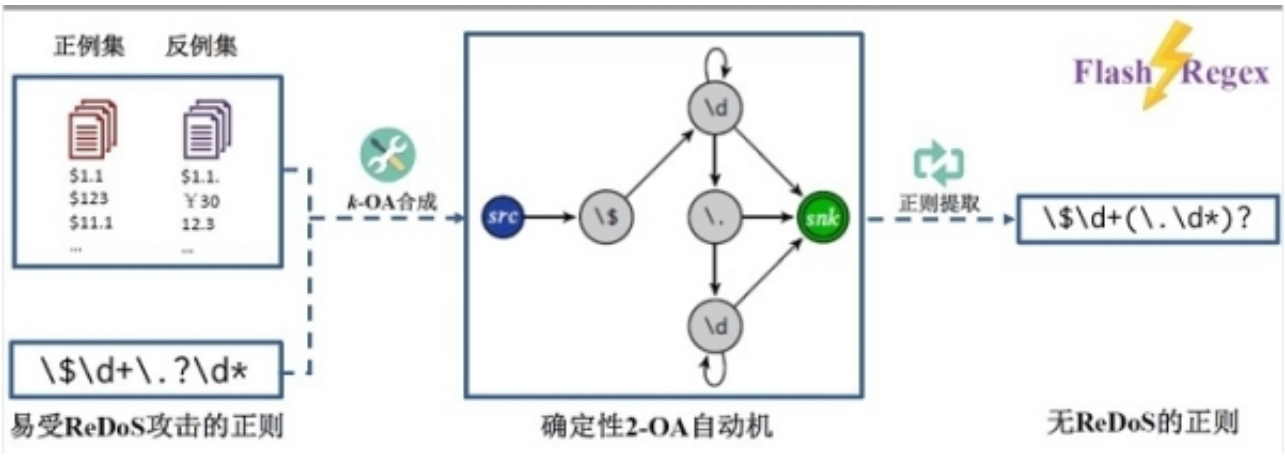
软件所博士生李贞霆为上述两篇论文的第一作者，陈海明为上述两篇论文的通讯作者。

No.	Pattern	Vuln. Degree	Vuln. Position	Attack String	Validated
#1	NQ	Exponential	(\d+)+	'a\n\ta' + '1' × 30 + '!'	√
#2	EOD	Exponential	(\s|\t)+	'a\n\ta1' + '\t' × 30 + '!'	√
#3	EOA	Exponential	(a*b+a*)+	'a\n\ta1\t' + 'ba' × 30 + '!'	√
#4	POA	Polynomial	\s+\s+	'a' + '\t\t' × 10000 + '!'	√
#5	SLQ	Polynomial	a+	'a' × 10000 + '!'	√

ReDoSHunter工具检测流程实例

Regex Engine	Java-8					Java-13					Python-3.7					Node.js-14				
Technique	TP	FP	FN	Prec (%)	Rec (%)	TP	FP	FN	Prec (%)	Rec (%)	TP	FP	FN	Prec (%)	Rec (%)	TP	FP	FN	Prec (%)	Rec (%)
RXXR2	224	5	10,121	97.82	2.17	216	13	10,032	94.32	2.11	213	16	9,594	93.01	2.17	219	10	9,427	95.63	2.27
Rexploiter	2,052	288	8,293	87.69	19.84	2,041	299	8,207	87.22	19.92	1,955	385	7,852	83.55	19.93	1,915	425	7,731	81.84	19.85
NFAA	975	13	9,370	98.68	9.42	968	20	9,280	97.98	9.45	857	131	8,950	86.74	8.74	842	146	8,804	85.22	8.73
safe-regex	3,760	2,348	6,585	61.56	36.35	3,715	2,393	6,533	60.82	36.25	3,586	2,522	6,221	58.71	36.57	3,540	2,568	6,106	57.96	36.70
Regexploit	1,051	2	9,294	99.81	10.16	1,051	2	9,197	99.81	10.26	1,044	9	8,763	99.15	10.65	1,032	21	8,614	98.01	10.70
SDL	112	0	10,233	100	1.08	108	4	10,140	96.43	1.05	98	14	9,709	87.50	1.00	102	10	9,544	91.07	1.06
ReScue	188	0	10,157	100	1.82	183	5	10,065	97.34	1.79	175	13	9,632	93.09	1.78	179	9	9,467	95.21	1.86
ReDoSHunter	**10,345**	**0**	**0**	**100**	**100**	**10,248**	**0**	**0**	**100**	**100**	**9,807**	**0**	**0**	**100**	**100**	**9,646**	**0**	**0**	**100**	**100**
Real Vulnerabilities	10,345					10,248					9,807					9,646				

ReDoS漏洞检测工具在三大数据集上的识别效果比较



FlashRegex工具修复流程

研究团队单位：软件研究所

更多 科学进展 请访问 https://www.iikx.com/news/progress/