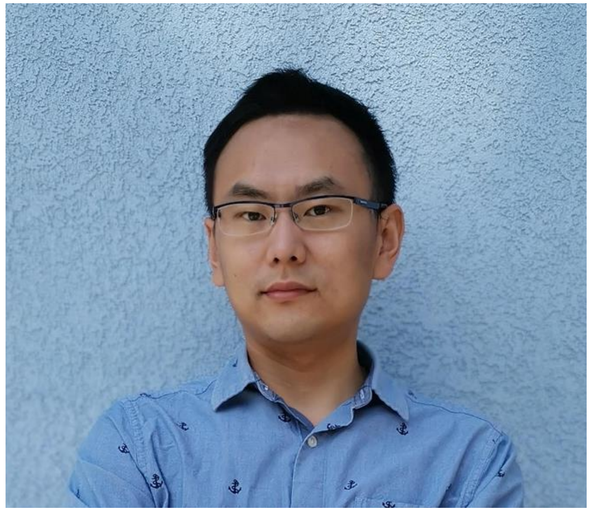
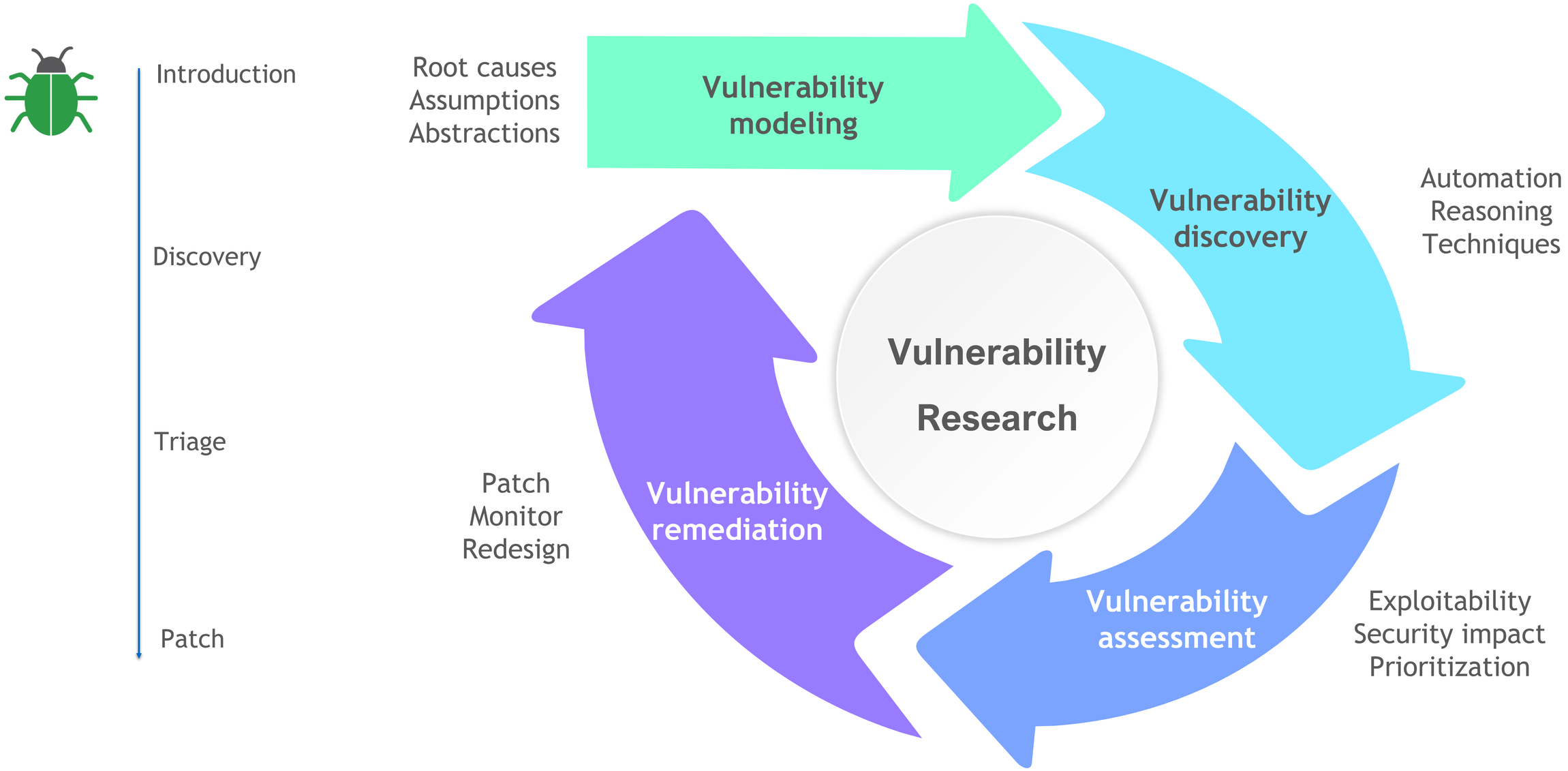
Self intro

- Zhiyun Qian, CSE Professor
- Web: http://www.cs.ucr.edu/~zhiyunq
- Email: zhiyunq@cs.ucr.edu
- Area: System and network security, vulnerability research
- Techniques: program analysis, reverse engineering, fuzzing, model checking, and AI / machine learning

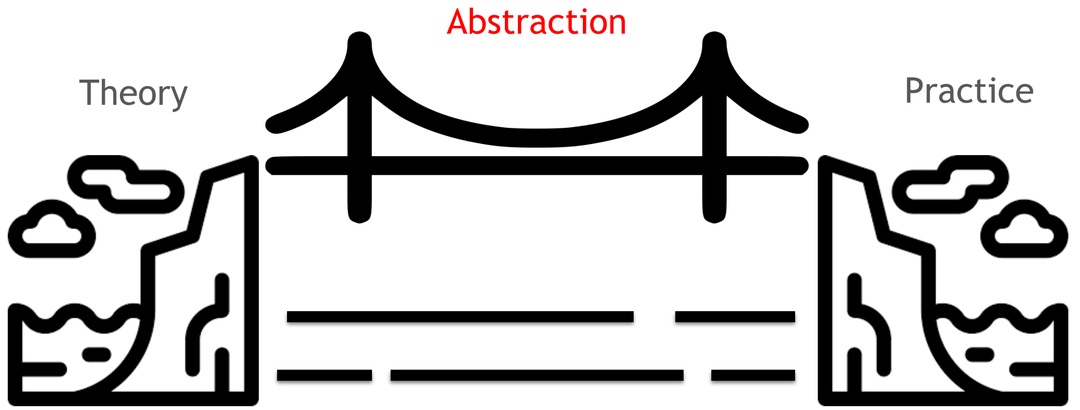


Vulnerability Lifecycle and Vulnerability Research



Bridging the Gap (academia vs. industry)

“In theory there is no difference between theory and practice. In practice there is.”



Security properties (e.g., non-interference)
Formal methods (e.g., PL)
…

Broken assumptions
Diverse domains
Messy details
Arbitrary constraints
…

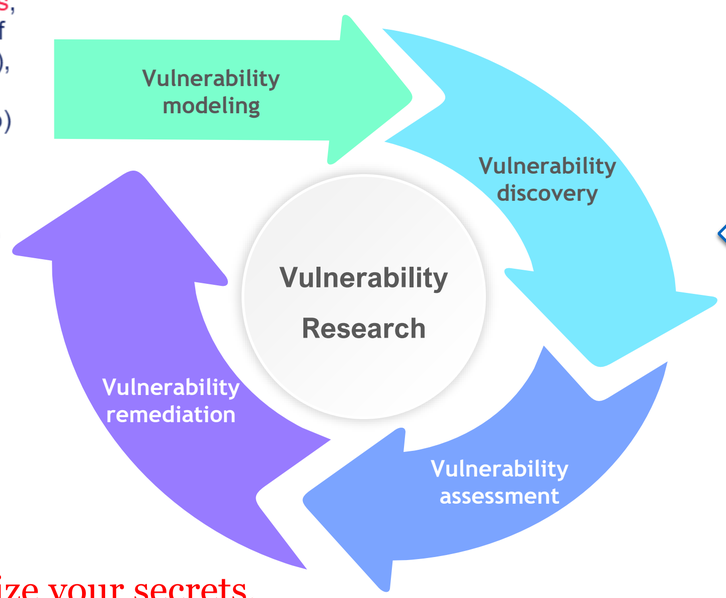
Vulnerability Research

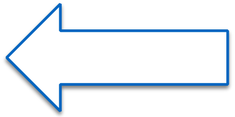
ACM CCS 2020

Distinguished Paper Award

- DNS Cache Poisoning Attack Reloaded: Revolutions With Side Channels, Keyu Man (University of California, Riverside), Zhiyun Qian (University of California, Riverside), Zhongjie Wang (University of California, Riverside), Xiaofeng Zheng (Qi-AnXin Group, Tsinghua University), Youjun Huang (Tsinghua University), Haixin Duan (Tsinghua University, Qi-AnXin Group)

Academia





Hacking community

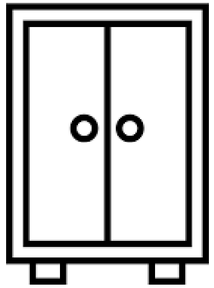
Internet Research Task Force

Applied Networking Research Prize

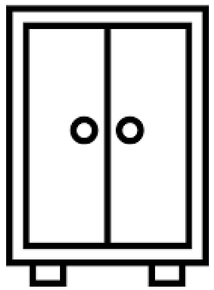
Off-path TCP exploit: how wireless routers can jeopardize your secrets,
Weiteng Chen and Zhiyun Qian

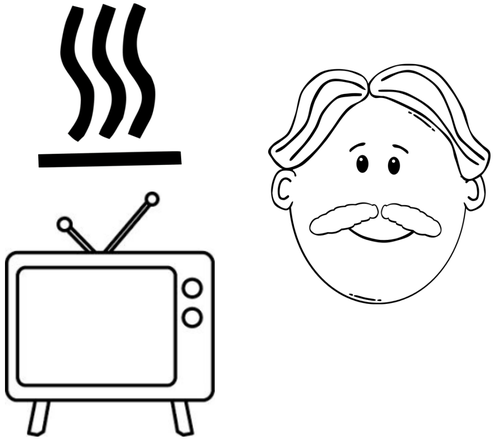
Usable automation tools

Modeling Side Channel Vulnerabilities

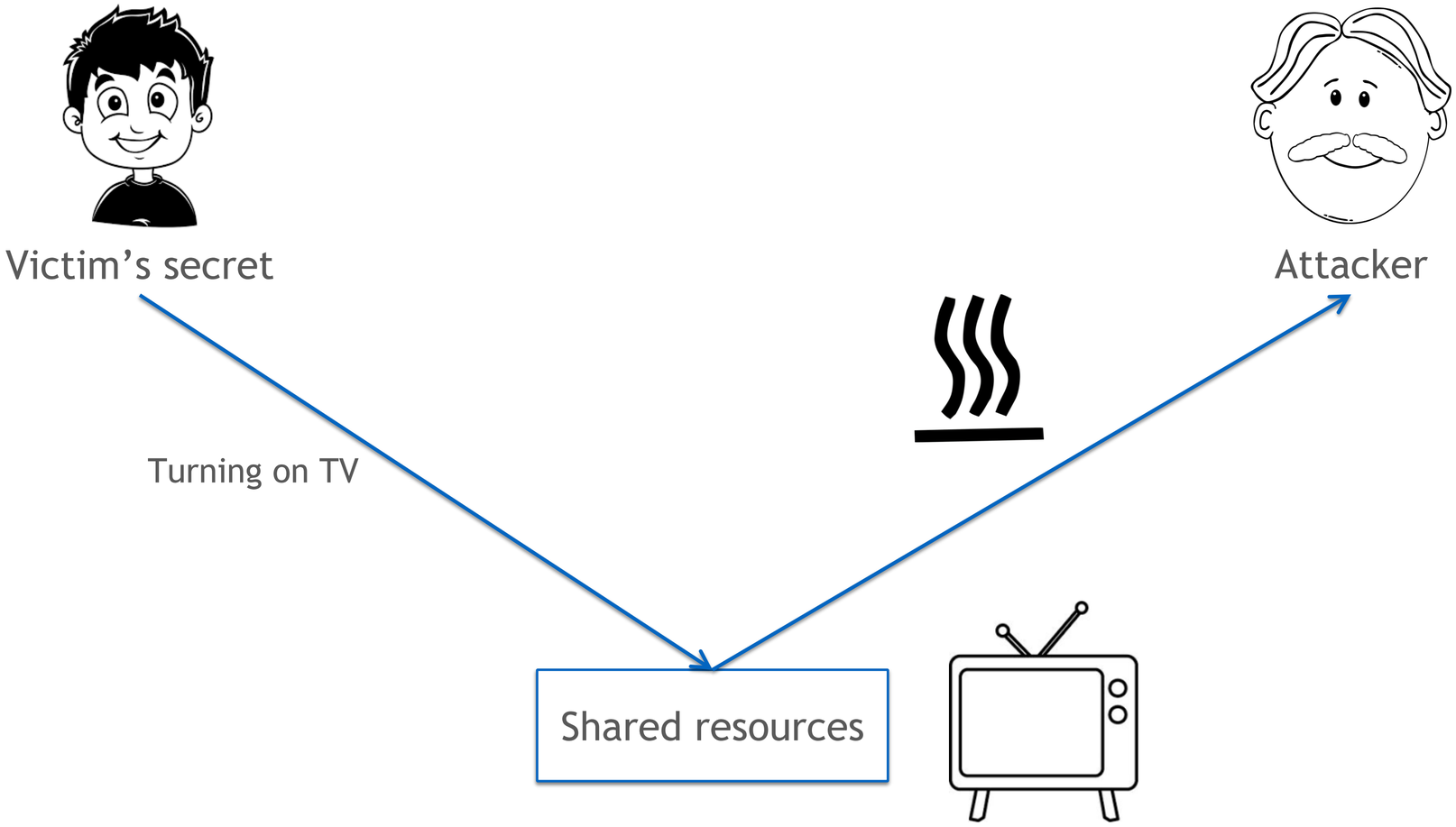


Modeling Side Channel Vulnerabilities



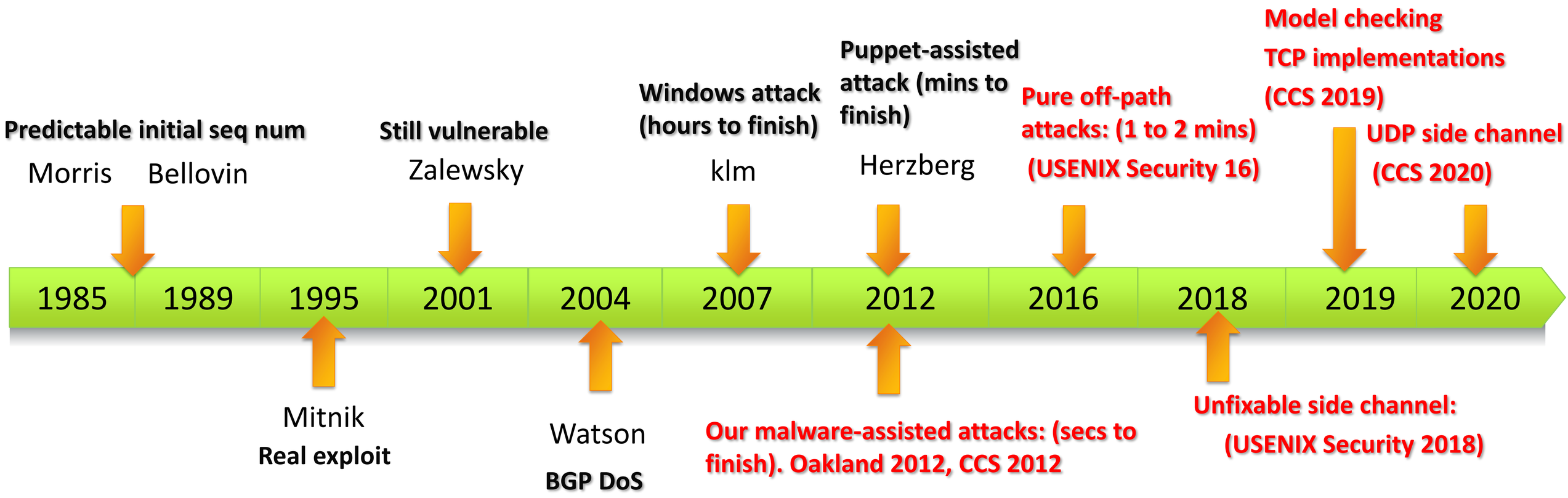


Modeling Side Channel Vulnerabilities

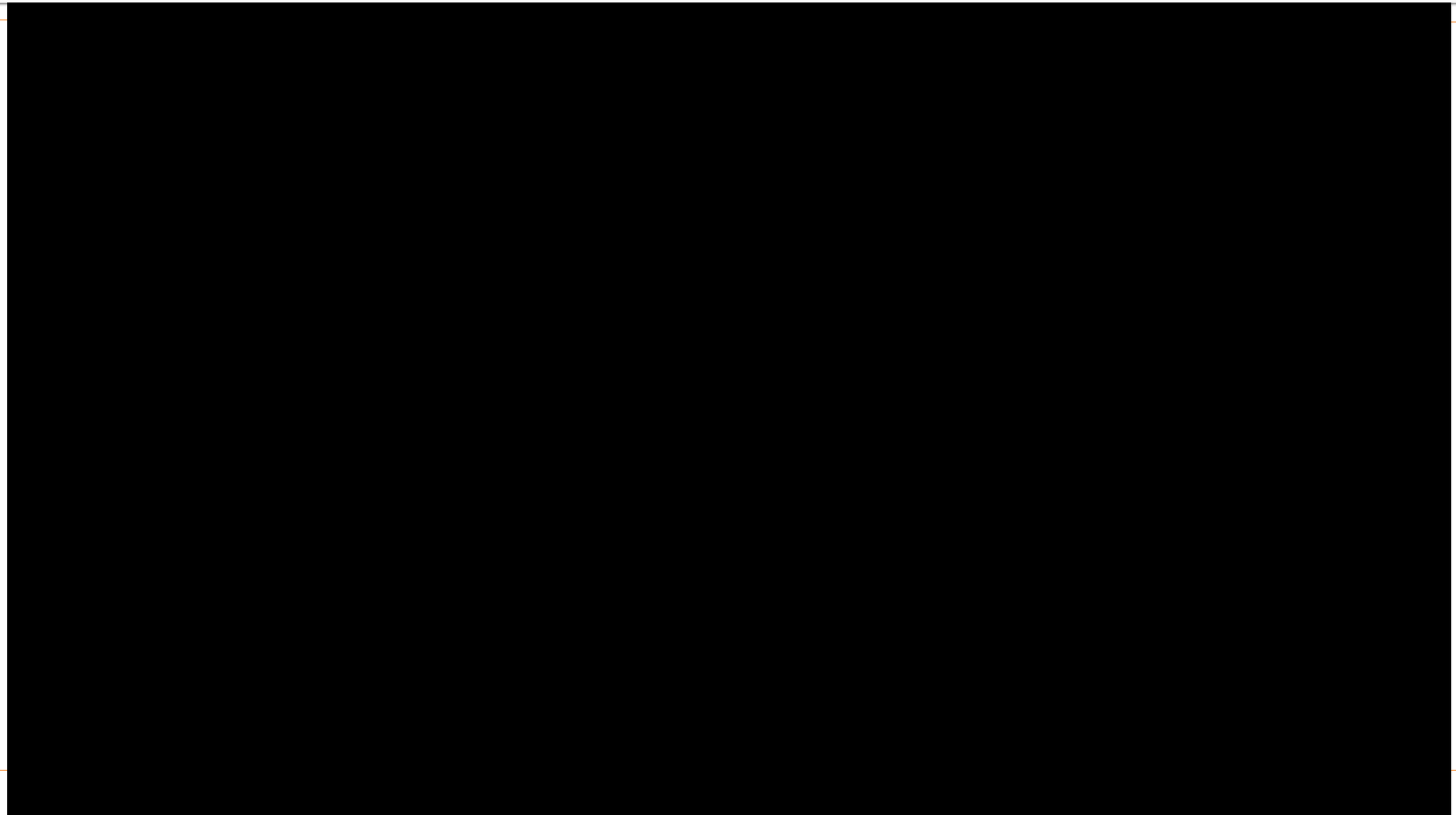


Network Protocol Side Channels

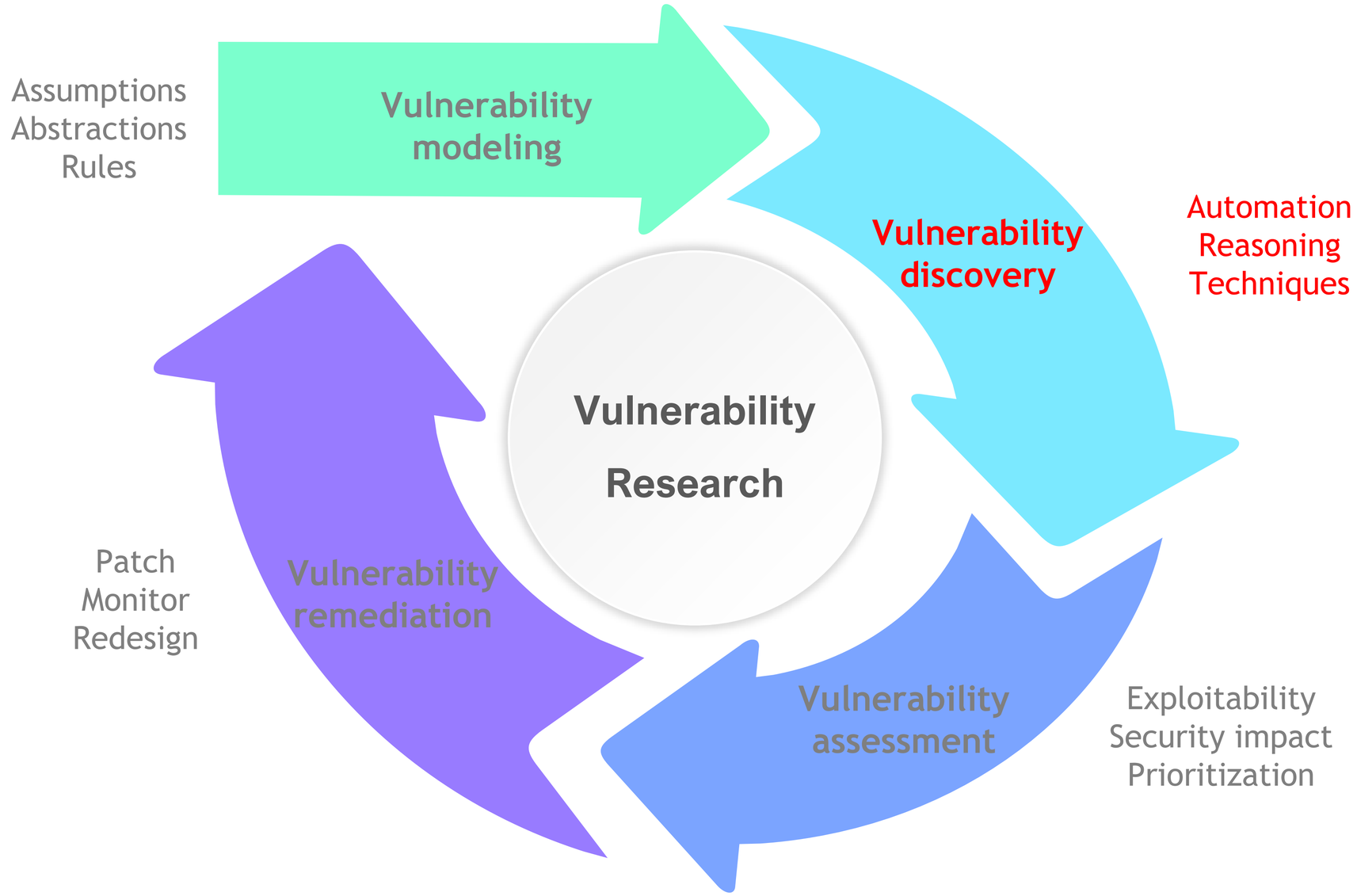
- History of off-path network attacks (side channels in TCP and UDP)



Network Protocol Side Channels



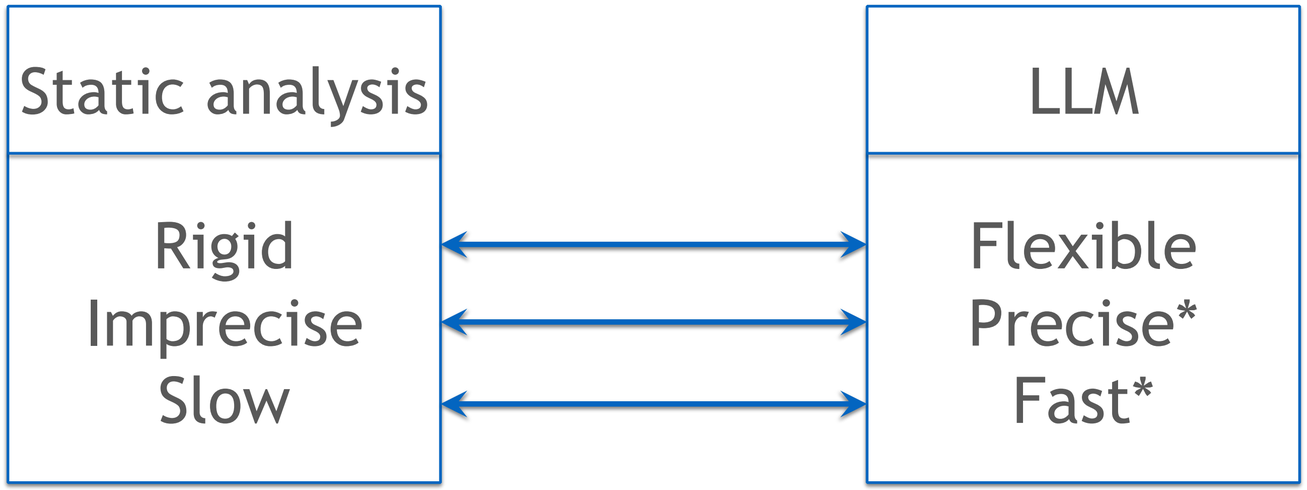
Vulnerability Research Cycle



Program Reasoning Tools

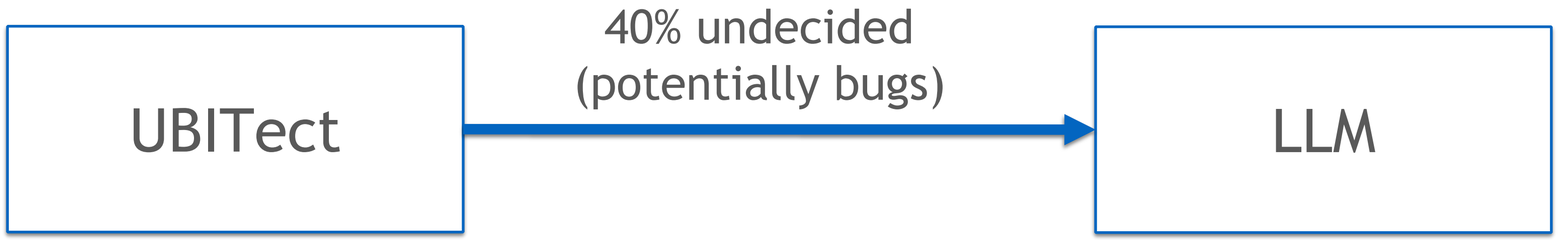
LLM-Assisted Program Analysis

- Static analysis + LLM > each alone



Example

- Complement a published static analysis tool (UBITect) with LLM:



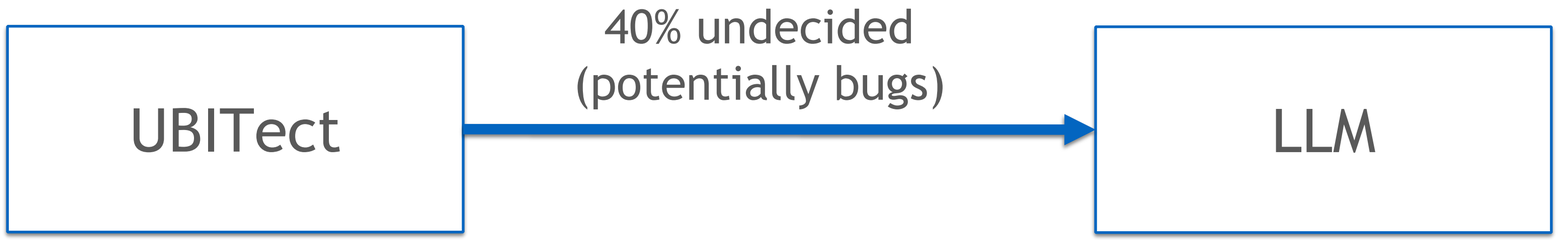

```
static int libcfs_ip_str2addr(...){
  unsigned int   a, b, c, d;
  if (sscanf(str, "%u.%u.%u.%u%n",
        &a, &b, &c, &d, &n) >= 4 && ...){
      // use of a, b, c, d
    }
}
```

Are these variables used without initialization?

Answer: No

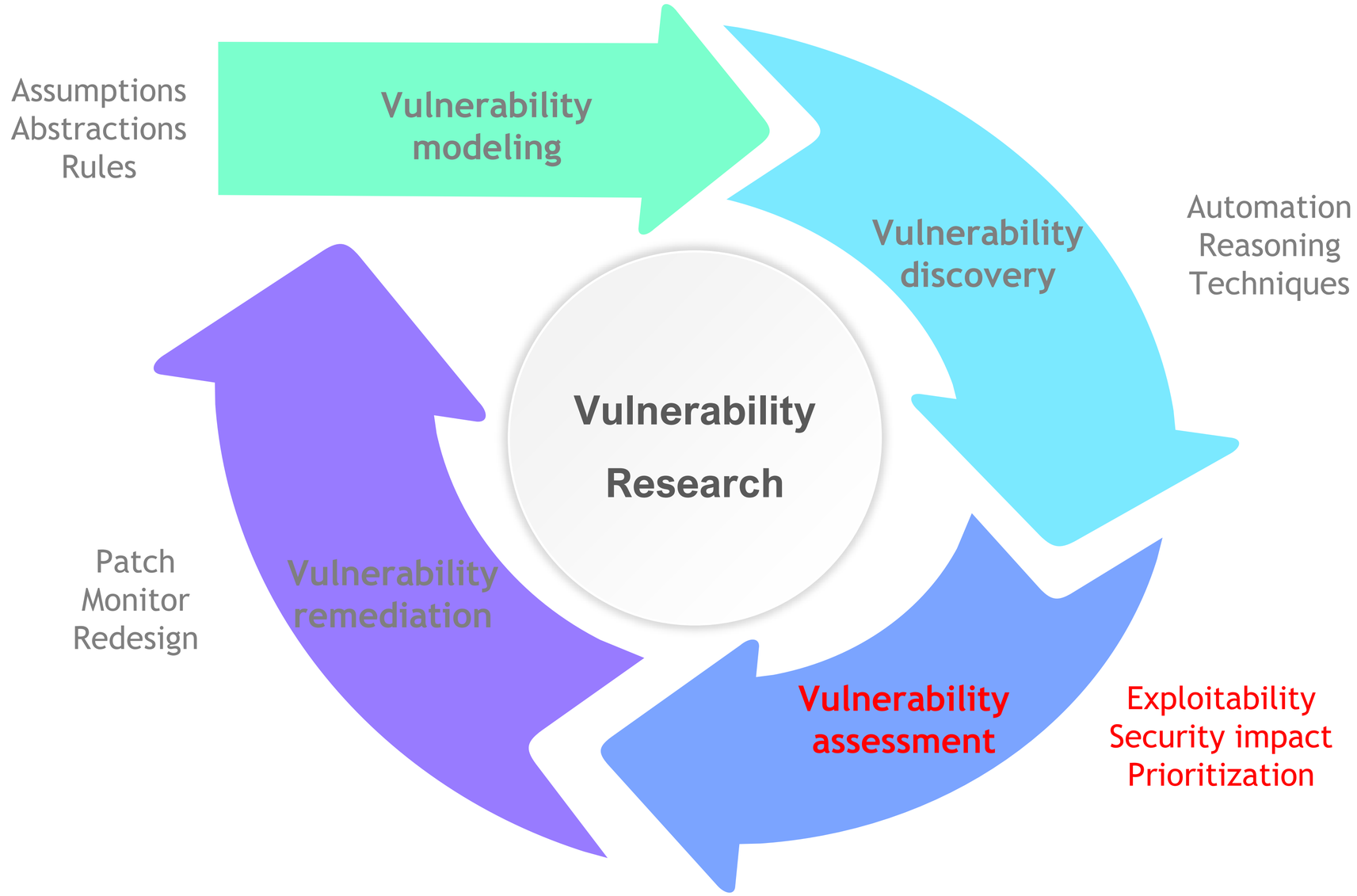
Example

- Complement a published static analysis tool (UBITect) with LLM:



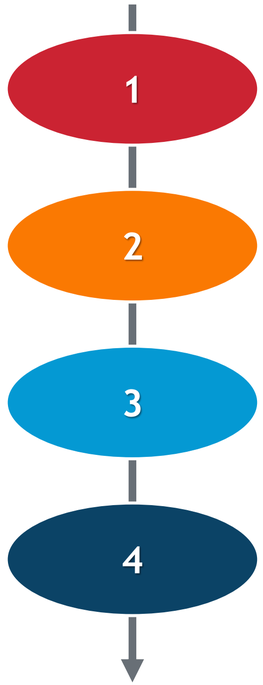
- Results
  - Filter **hundreds of false positives** produced by static analysis tools
  - Found **12 unknown vulnerabilities**

Vulnerability Research Cycle



Linux Kernel Bug Triage Pipeline

Fuzzer-exposed bugs
in Linux upstream



Apply to downstream kernels?

Require privilege to trigger?

High-risk primitives?

Exploitable?

(Multiple top-tier papers and tools published)

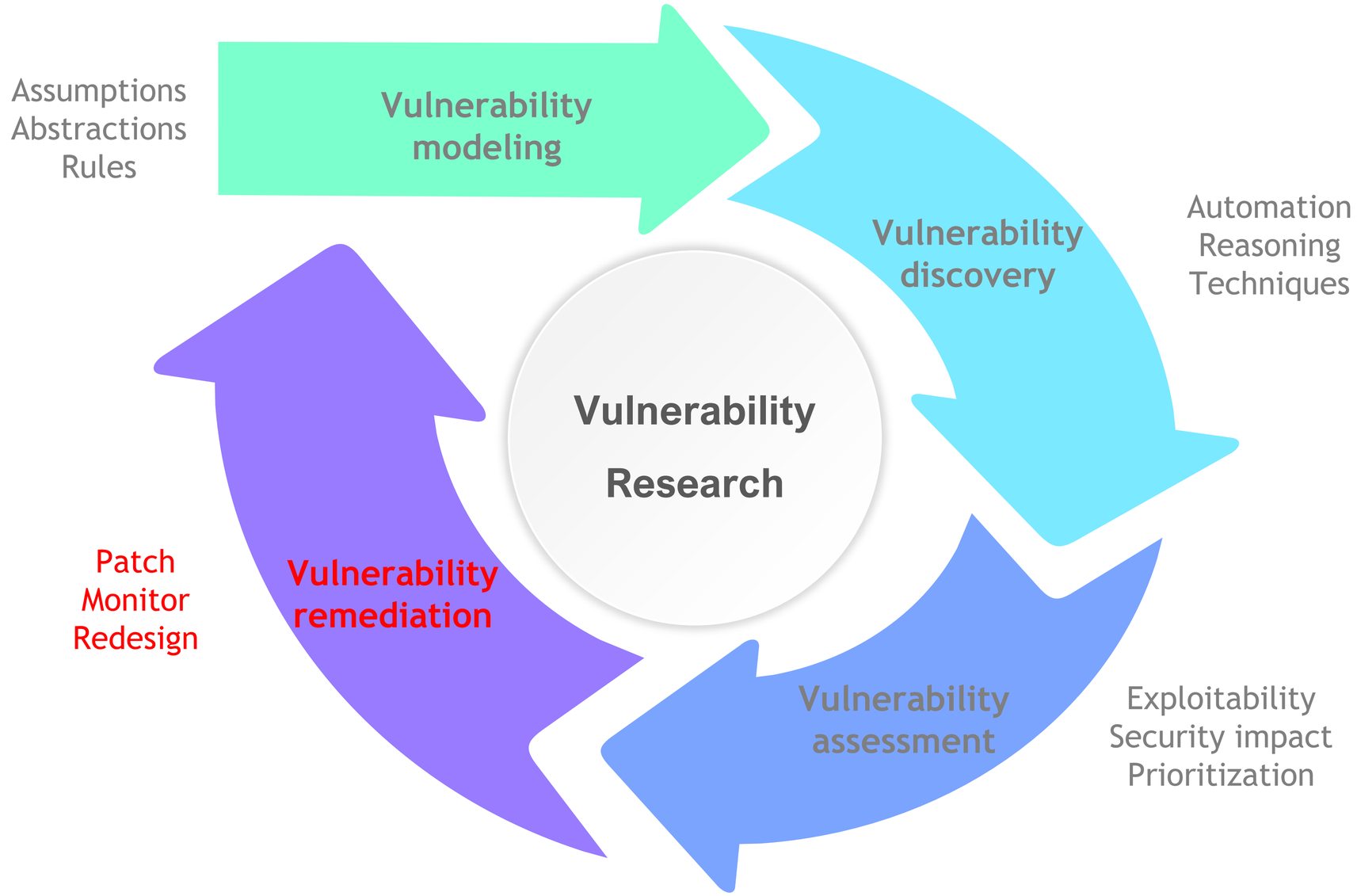
Example Results

- 53 likely exploitable bugs from 1,000+ public bugs

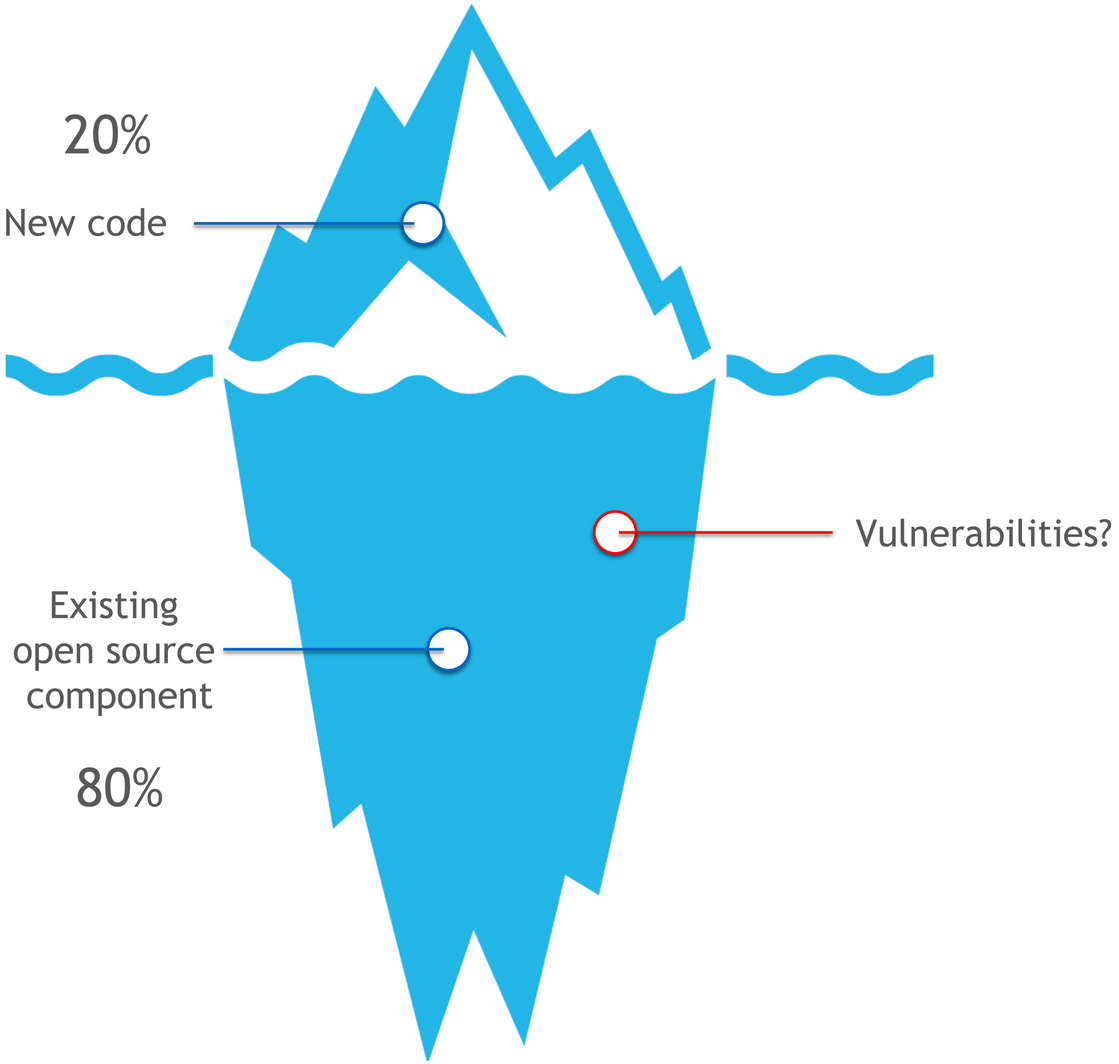
Bug	Bug Primitive	Affect	Before Adaptation		Environment Adaptation			Privilege Adaptation		After Adaptation	
			root	normal	EA1	EA2	EA3	PA1	PA2	root	normal
CVE-2022-27666*	OOB W	UFD	F	-	-	UD	UD	UD	UFD	-	UFD
CVE-2022-0185	OOB W	UFD	UFD	-	-	-	-	-	UFD	-	UFD
CVE-2021-22600	DF	UFD	UFD	-	-	-	-	-	UFD	-	UFD
CVE-2021-22555	OOB W	UFD	F	-	-	UD	UD	UD	UFD	-	UFD
CVE-2021-4154	CFH	UFD	UFD	-	-	-	-	-	UFD	-	UFD
CVE-2021-3715	UAF W	U	U	-	-	-	-	-	U	-	U
cf7393b*	UAF W	UFD	UFD	-	-	-	-	-	UF	D	UF
4b0830a	UAF W	UFD	U	D	-	-	F	-	F	U	FD
e67f2fc	UAF W	UFD	-	-	-	UFD	-	-	UFD	-	UFD
2389bfc*	CFH	UFD	UFD	-	-	-	-	-	F	UD	F
403eb21	CFH	UFS	UF	-	S	-	-	-	UF	S	UF
f4c90f2*	OOB W	UFDS	UFDS	-	-	-	-	-	UFS	D	UFS
380acd1*	DF	UF	UF	-	-	-	-	-	UF	-	UF
b53aed2	OOB W	UFD	UFD	-	-	-	-	-	UFD	-	UFD
e2d0f38	CFH	F	-	-	-	F	-	-	F	-	F
d35e6e8	NPD W	UFD	UFD	-	-	-	-	-	UFD	-	UFD
60e3243	CVW	UFDS	UFDS	-	-	-	-	-	UFDS	-	UFDS
a53b68e	OOB W	UF	F	-	-	U	U	U	UF	-	UF
5ad0e07	NPD W	UF	UF	-	-	-	-	-	UF	-	UF
7c7245f	OOB W	UFD	UFD	-	-	-	-	-	UFD	-	UFD
f1834e1	CFH	FD	-	-	D	FD	-	-	-	-	FD
ed87cd6	CFH	F	F	-	-	-	-	-	F	-	F
e4c5c37	AVW	FD	-	-	-	FD	-	-	F	D	F
e3e31b1	UAF W	FD	F	-	-	D	-	-	F	D	F
b8febdb	UAF W	FD	-	-	-	-	FD	FD	FD	-	FD
ba1aecb	DF	FD	F	-	-	D	-	-	FD	-	FD
955089c	CFH	FD	FD	-	-	-	-	-	FD	-	FD
457491c	CFH	UFD	-	-	UFD	-	UFD	UFD	UFD	-	UFD
6578348	CFH	UFDS	UD	FS	-	-	-	-	UD	-	UFDS
26de18d	CVW	UFD	-	-	U	UFD	-	-	UF	D	UF
418578d	UAF W	UFD	-	-	UD	UFD	-	-	F	-	UFD
2a62245	OOB W	F	F	-	-	-	-	-	F	-	F
232223b	UAF W	FS	FS	-	-	F	-	-	F	S	F
27934d2	UAF W	UFDS	UFDS	-	-	-	-	-	UFS	D	UFS
26cb120	DF	UFD	UFD	-	-	-	-	-	UF	D	UF

Vulnerability Assessment → Automated bug triage

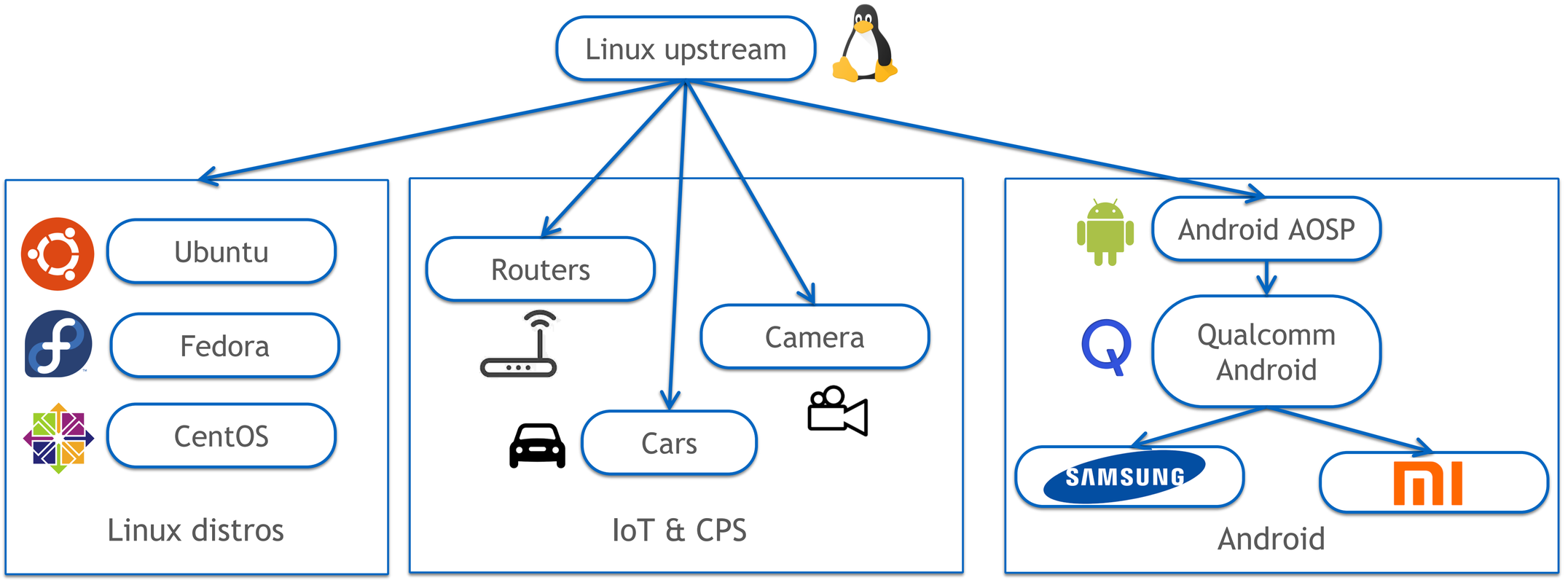
Vulnerability Research Cycle



Software Reuse and Supply Chain Security



Supply Chain Security: Case of Linux

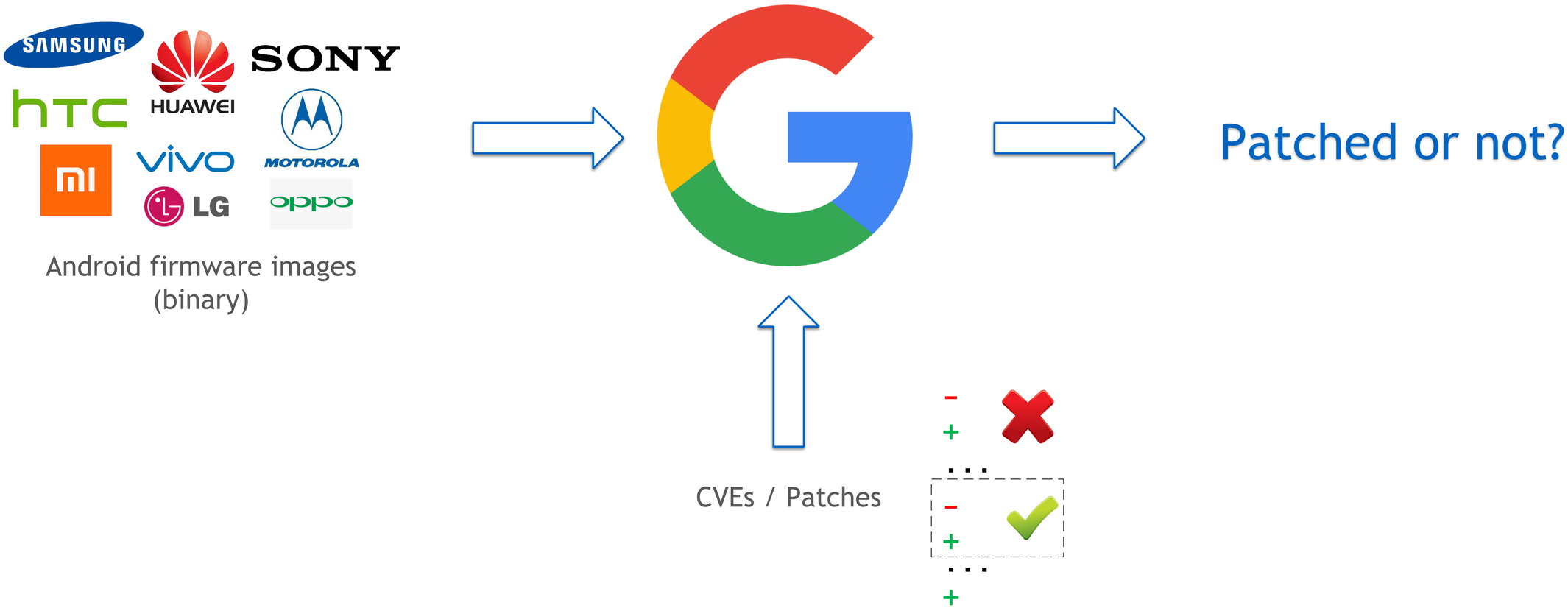


Not always open source!

- Binary only
- Snapshot only
- Significant delays

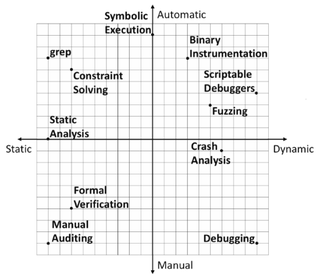
Patch Presence Test

- Google has no visibility into which OEM kernel images have patched a CVE

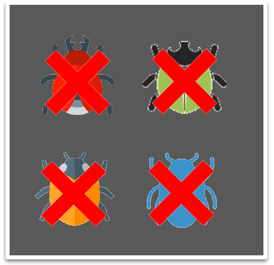


Conclusion

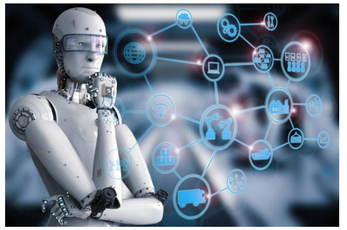
- Understanding novel types of vulnerabilities



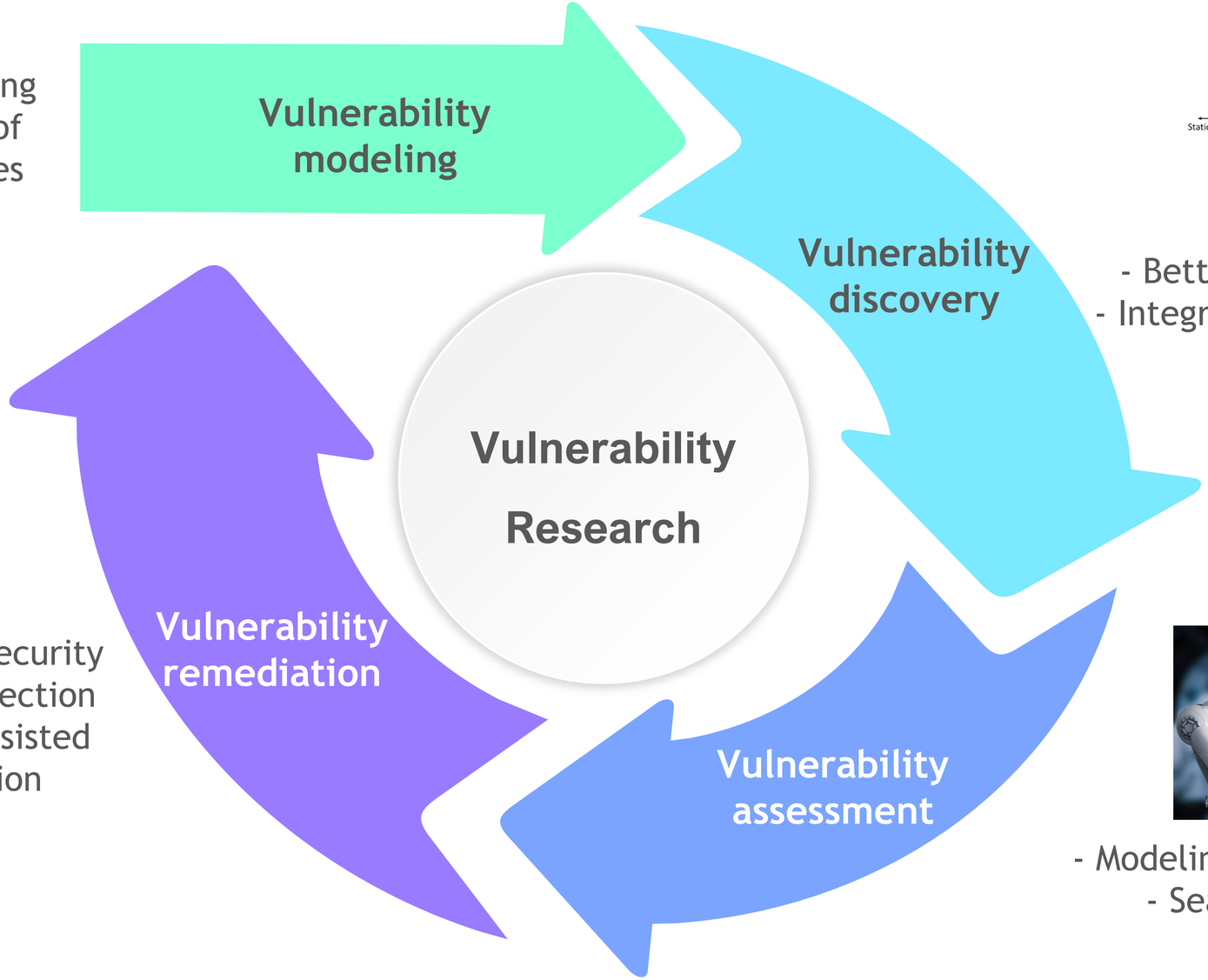
- Better reasoning tools
- Integration of techniques



- Supply chain security
- Selective protection
- Hardware-assisted remediation



- Modeling exploit techniques
- Search for impacts



https://github.com/seclab-ucr

End