Securing the Smart Home via a Two-Mode Security Framework

Devkishen Sisodia, Samuel Mergendahl, Jun Li, and Hasan Cam

August 8, 2018

SecureComm 2018

Network & Security
Research Laboratory

This project is in part the result of funding provided by the Science and Technology Directorate of the United States Department of Homeland Security under contract number D15PC00204. The views and conclusions contained herein are those of the authors and should not be interpreted necessarily representing the official policies or endorsements, either expressed or implied, of the Department of Homeland Security or the US Government.

Table of Contents

Table of Contents

IoT Security and Privacy at Home

- Internet of Things (IoT) supports various daily activities at home
  - Security cameras that monitor a home
  - Smart thermostats and lights for convenience
  - Motion sensors capable of telling when an elderly person has fallen
- However, IoT devices also present serious security and privacy concerns
- Addressing such concerns is critical to gaining the benefits of IoT

The Difficulty of Securing a Smart Home

- High level of heterogeneity
  - High-powered vs. low-powered
  - Plugged-in vs. battery-powered
  - Regulary updated vs. never updated
- Mesh networking
  - Central entity such as border router cannot monitor communication between devices


IEEE SPECTRUM

Opinion | Telecom | Security

20 Jun 2018 | 19:00 GMT

6 Reasons Why IoT Security Is Terrible

The Internet of Things bears little resemblance to traditional IT systems—and that makes it harder to protect

By **Stacey Higginbotham**

Dilemma

- Characteristics of IoT networks make it difficult to apply traditional security systems
- Many traditional security systems run heavy-weight security features, either continuously or periodically
  - Infeasible to run on many IoT devices
  - Negatively impact the network – effects are compounded in IoT networks

Our Proposition – TWINKLE

- Two-mode security framework
- Runs light-weight security features most of the time
- Only runs heavy-weight security features **on-demand**
- Goal: achieve equal accuracy as traditional security systems while consuming less resources

Table of Contents

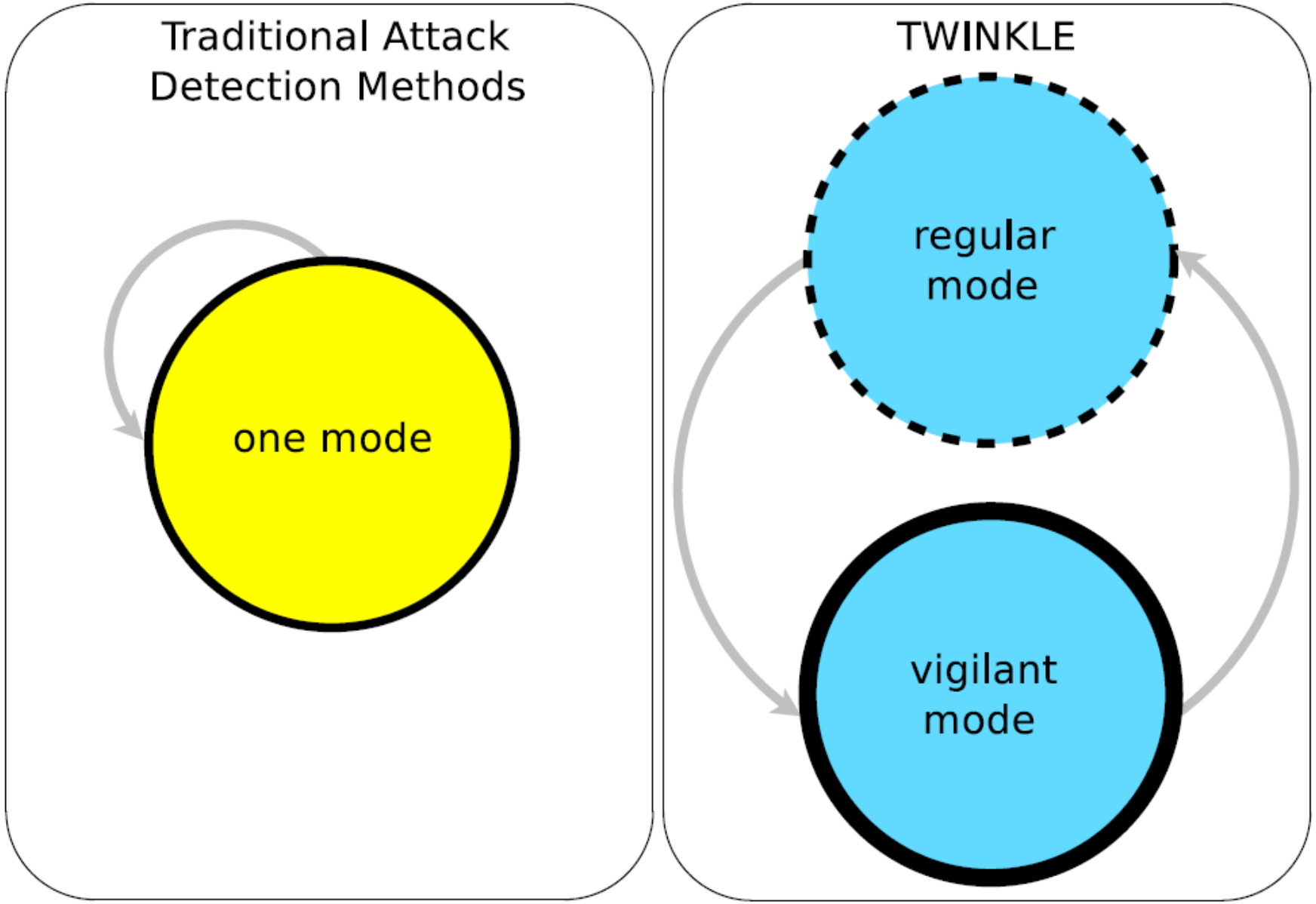
TWINKLE: A Two-Mode Security Framework for the Smart Home

- Address characteristics (and limitations) of IoT devices
- Designed to preserve salient features of classic security solutions
- Security applications (security solutions) can be plugged into TWINKLE
  - When plugged into TWINKLE, applications will run in two distinct modes: **regular mode** and **vigilant mode**

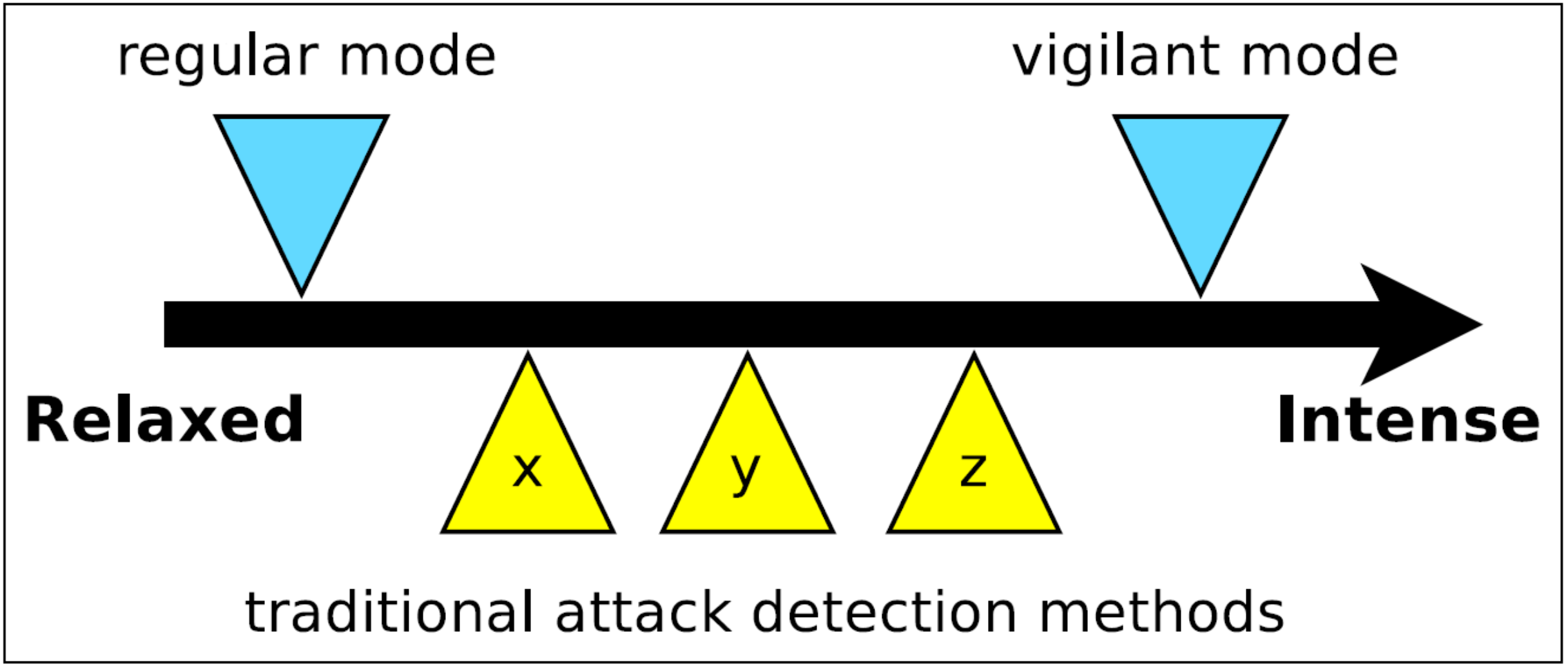
The Two Modes

- **Regular Mode**: run a lightweight algorithm to monitor network and raise an alarm whenever suspicious behavior is detected
- **Vigilant Mode**: run a more intensive algorithm to verify an attack and try to mitigate it

State Diagrams



Resource Consumption



Twinkle’s Three Components

Manager

Policy Checker

Watchdog

Manager

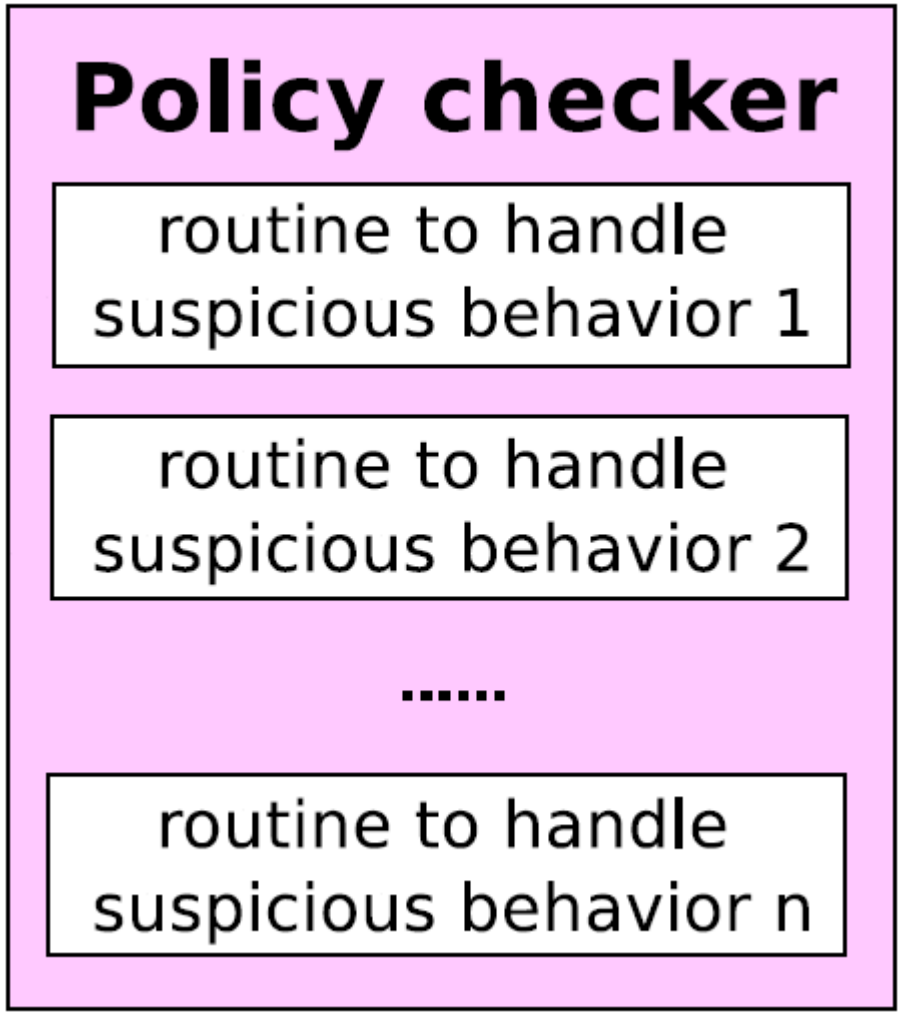
- Maintains information of the smart home network
- Supports a function to handle suspicious behavior, or the suspicious behavior handling function (SBHF)
- Maintains a suspicious behavior handling table (SBHT)

Suspicious behavior handling table (SBHT)

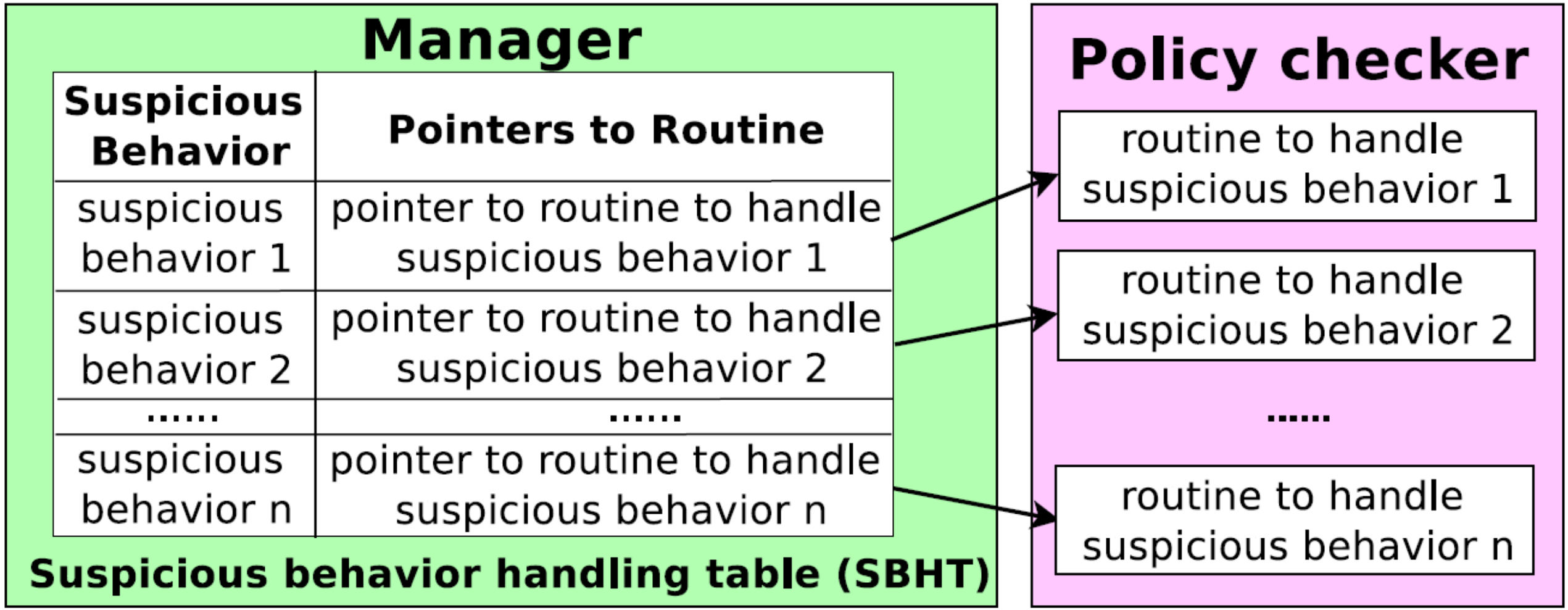
Suspicious Behavior	Pointers to Routine
suspicious behavior 1	pointer to routine to handle suspicious behavior 1
suspicious behavior 2	pointer to routine to handle suspicious behavior 2
.....	.....
suspicious behavior n	pointer to routine to handle suspicious behavior n

Policy Checker

- Maintains routines for handling suspicious behavior
- Such routines are usually heavyweight and should only be running in vigilant mode when invoked **on-demand**

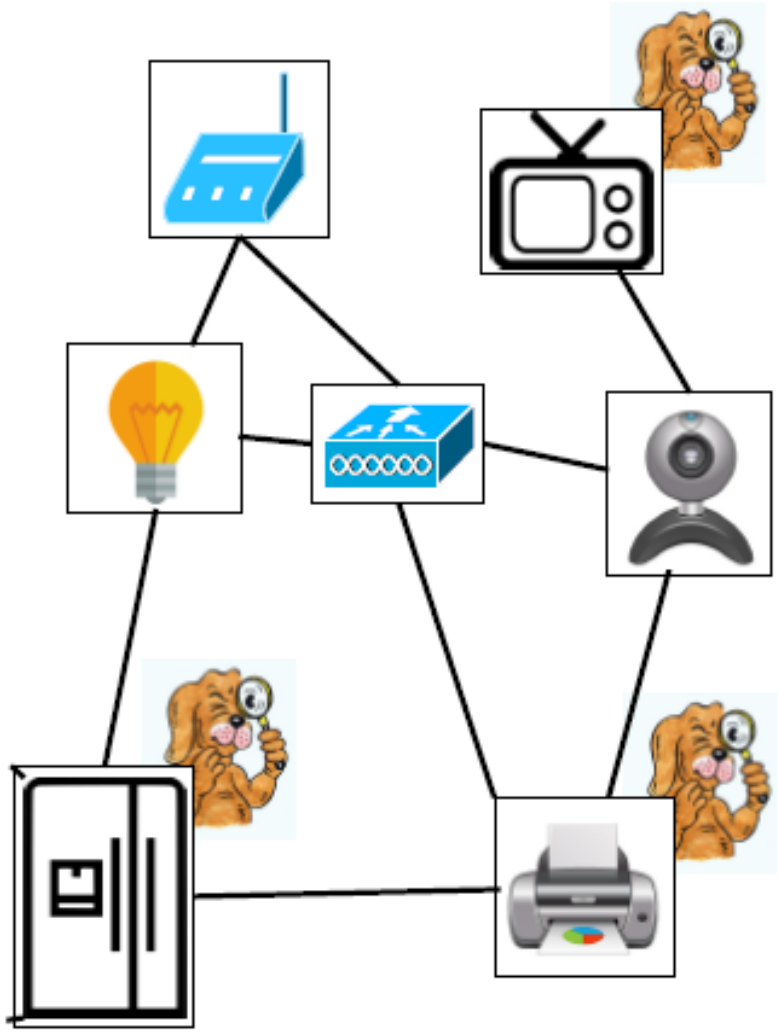


Interaction Between Manager and Policy Checker

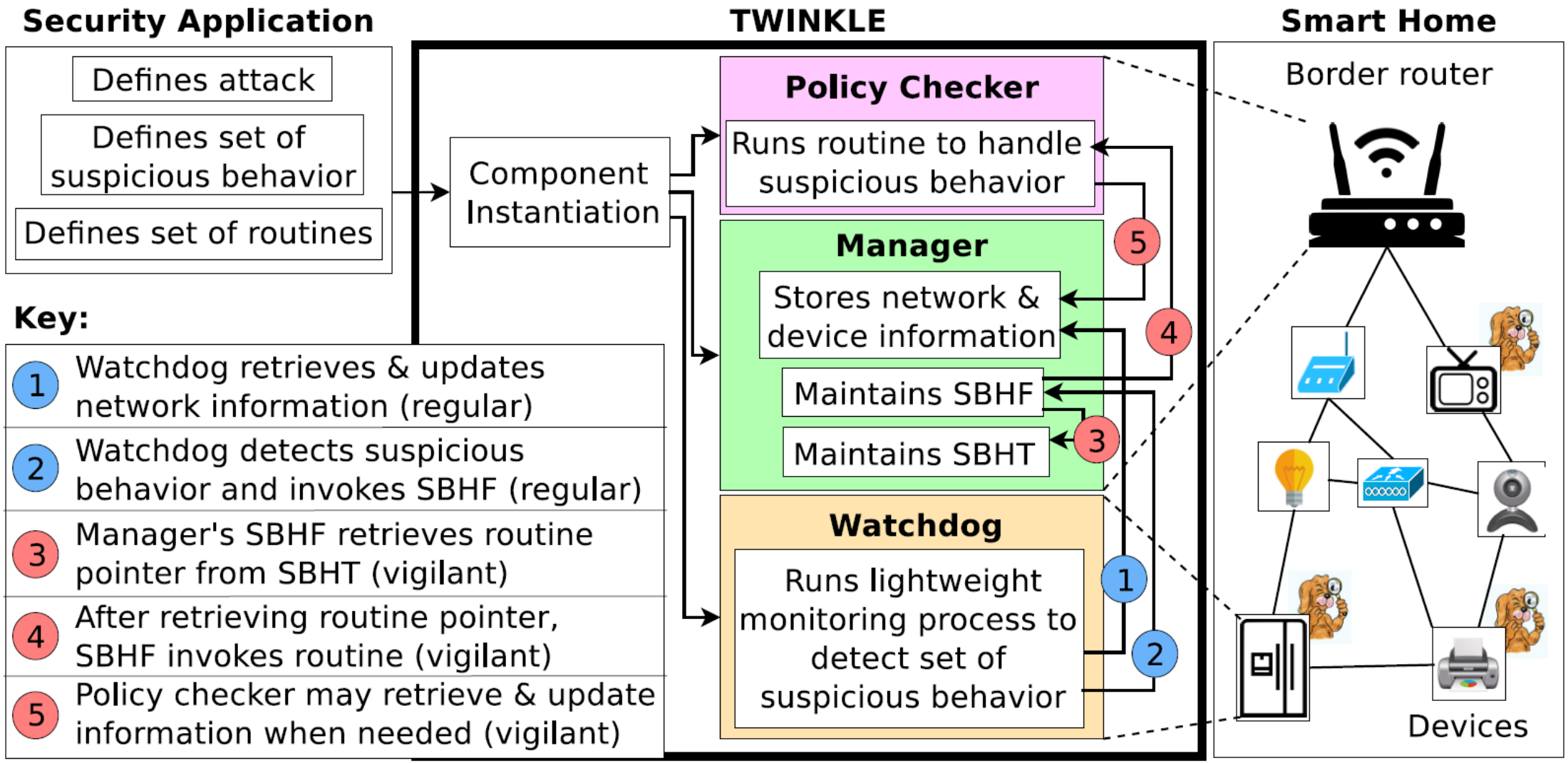


Watchdog

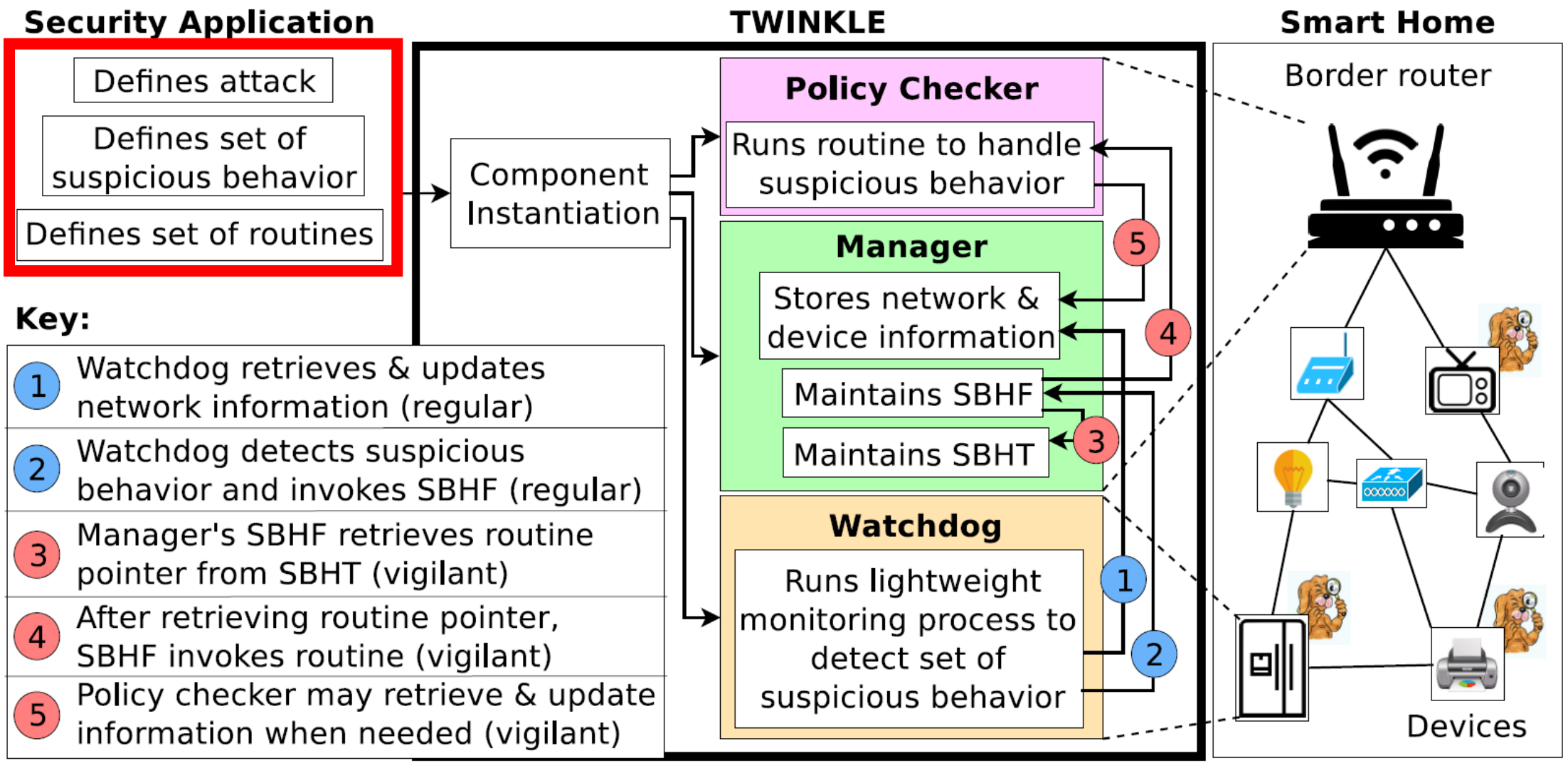
- Lightweight process that monitors smart home for suspicious behavior
- Multiple watchdogs can be running at multiple devices
- Whenever a watchdog detects suspicious behavior, it invokes SBHF to process the suspicious behavior
- As soon as function begins its execution, system will enter vigilant mode



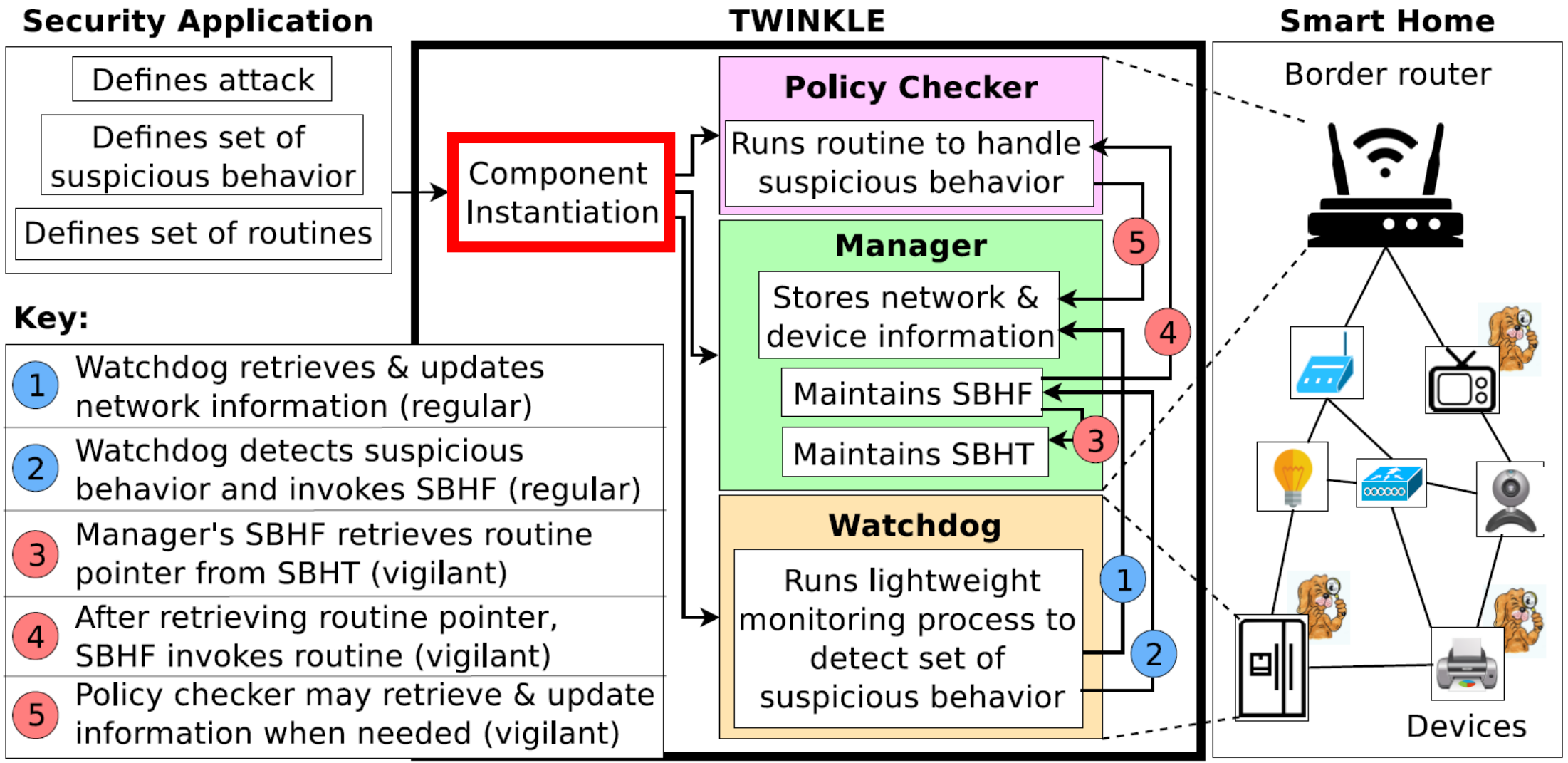
TWINKLE Architecture



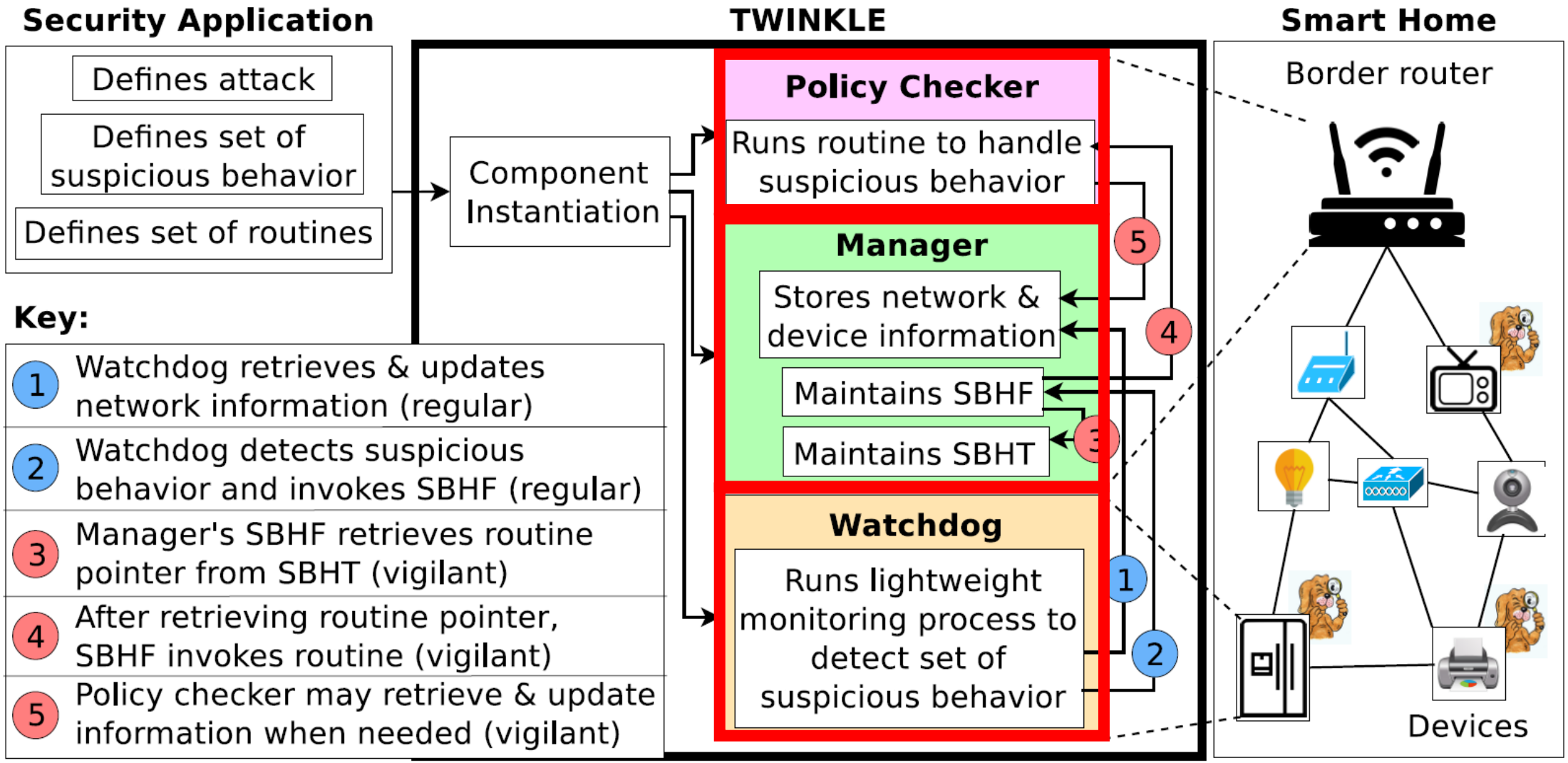
TWINKLE Architecture



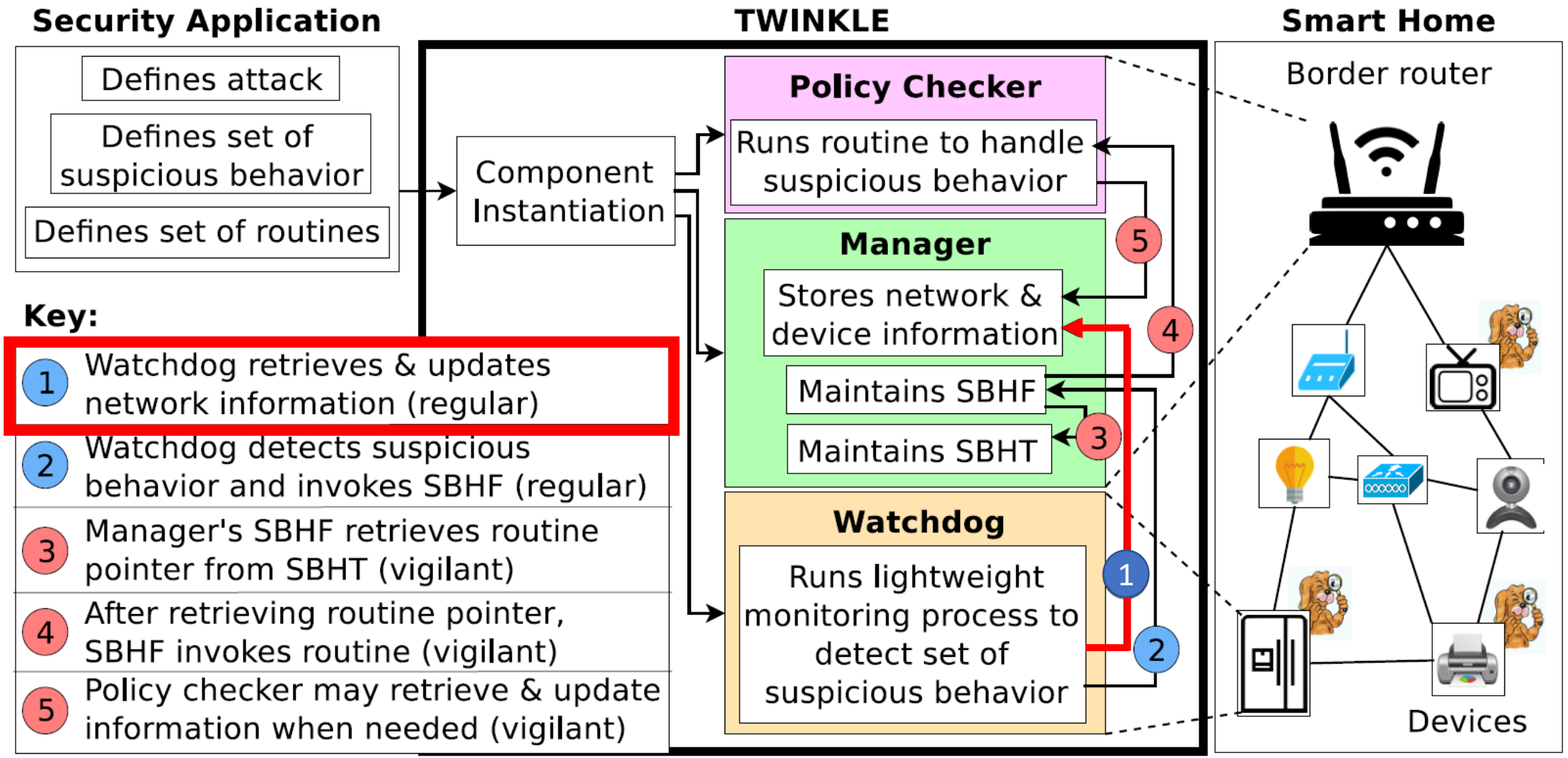
TWINKLE Architecture



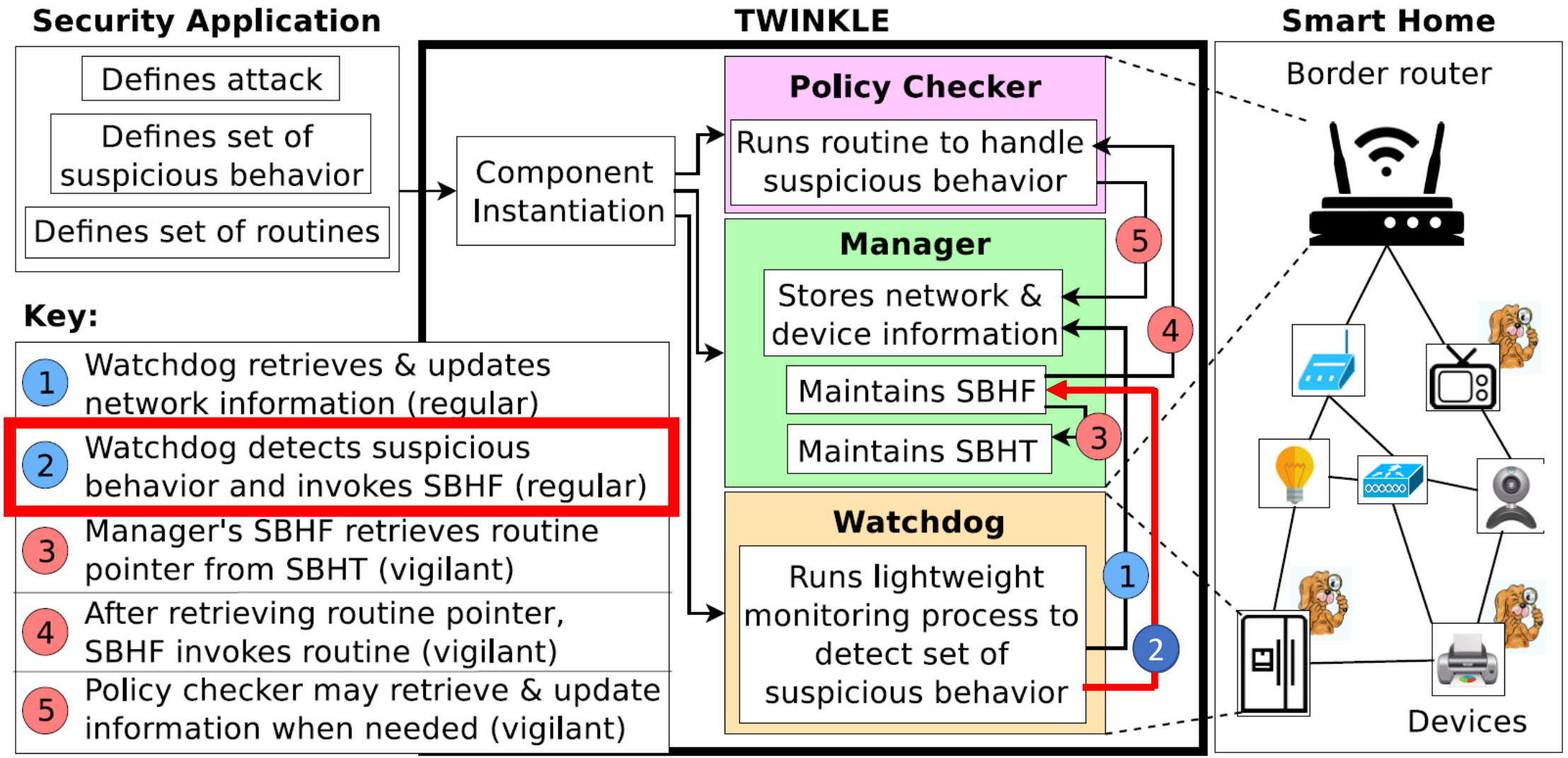
TWINKLE Architecture



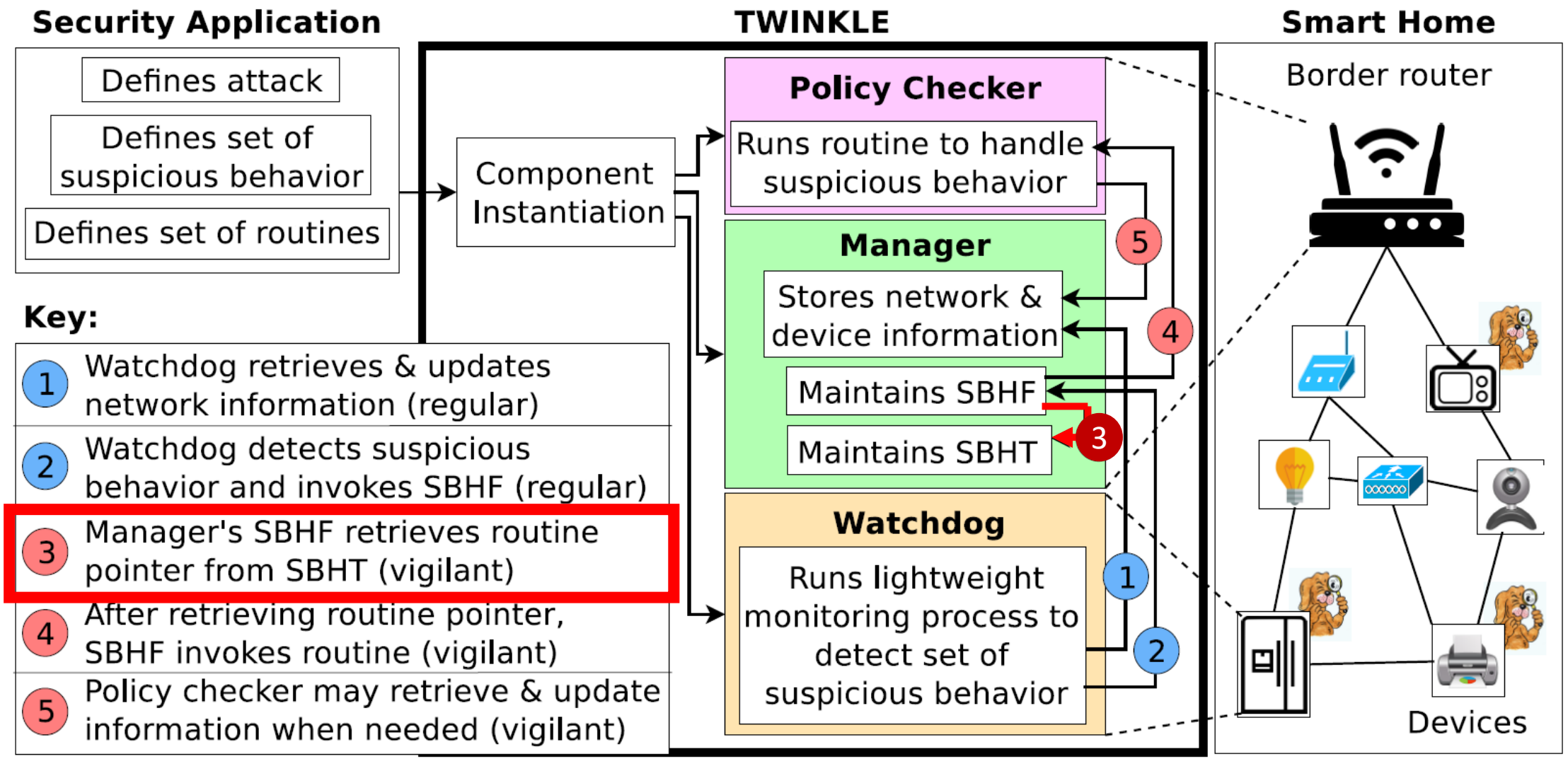
TWINKLE Architecture



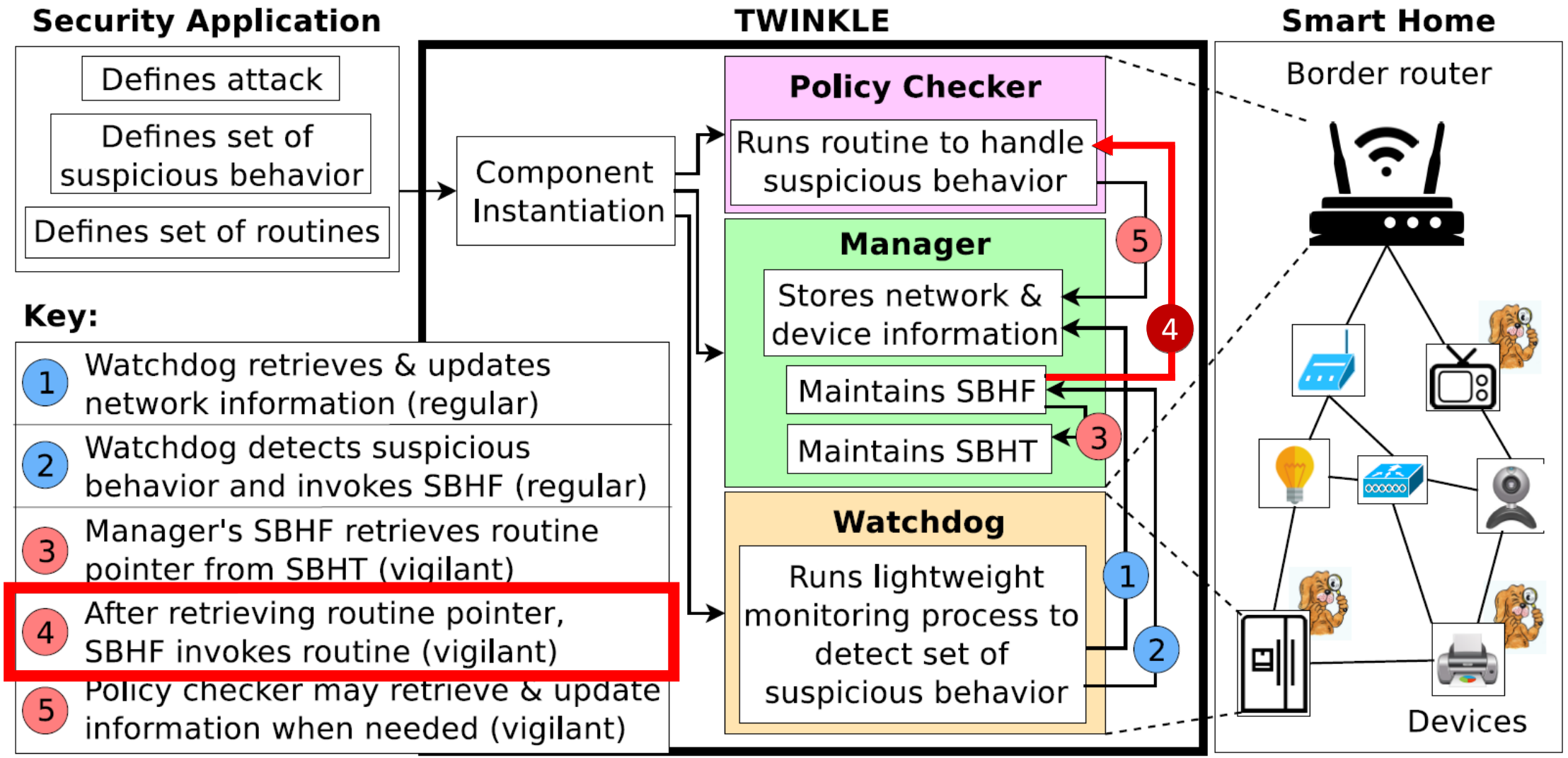
TWINKLE Architecture



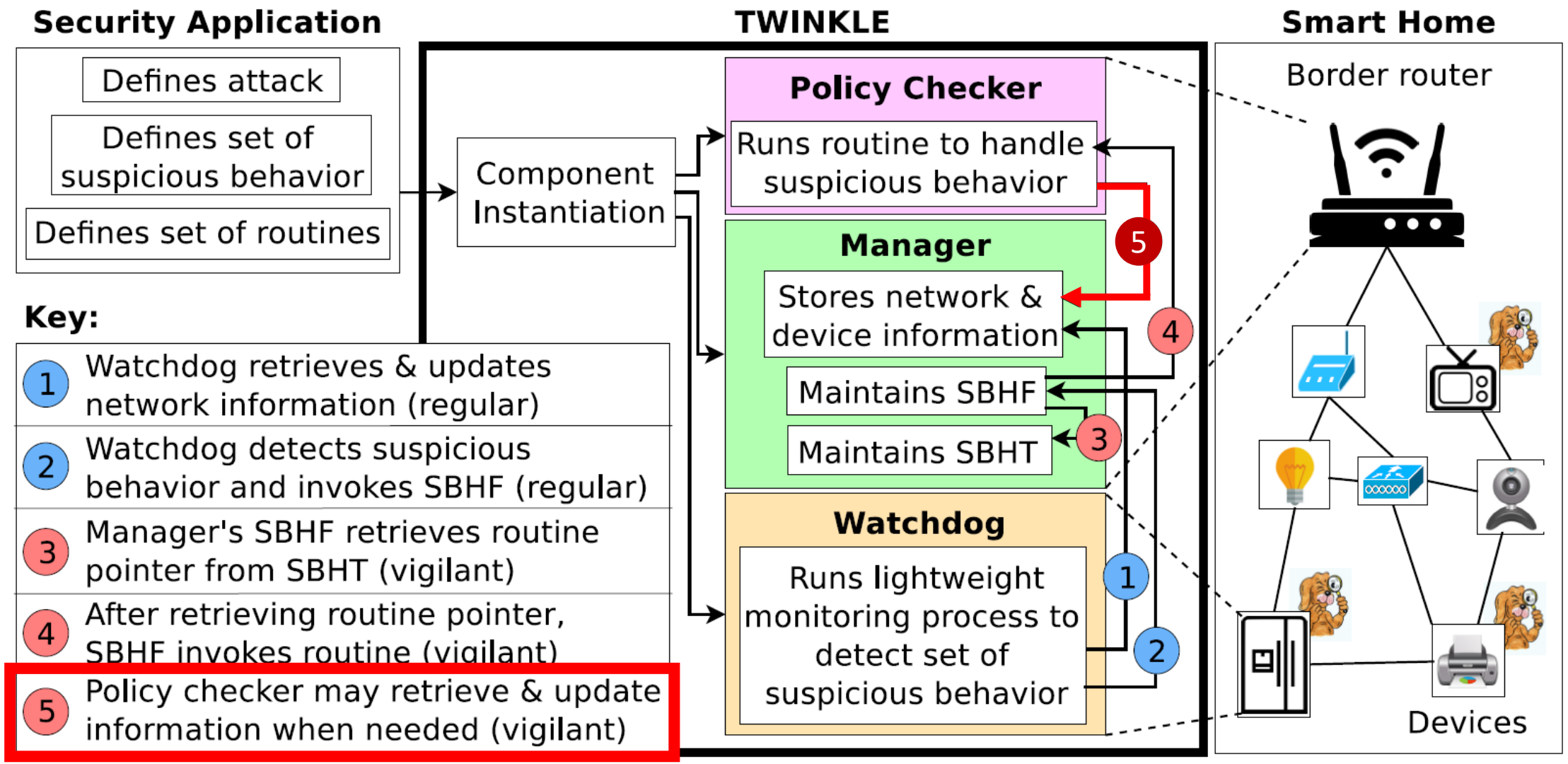
TWINKLE Architecture



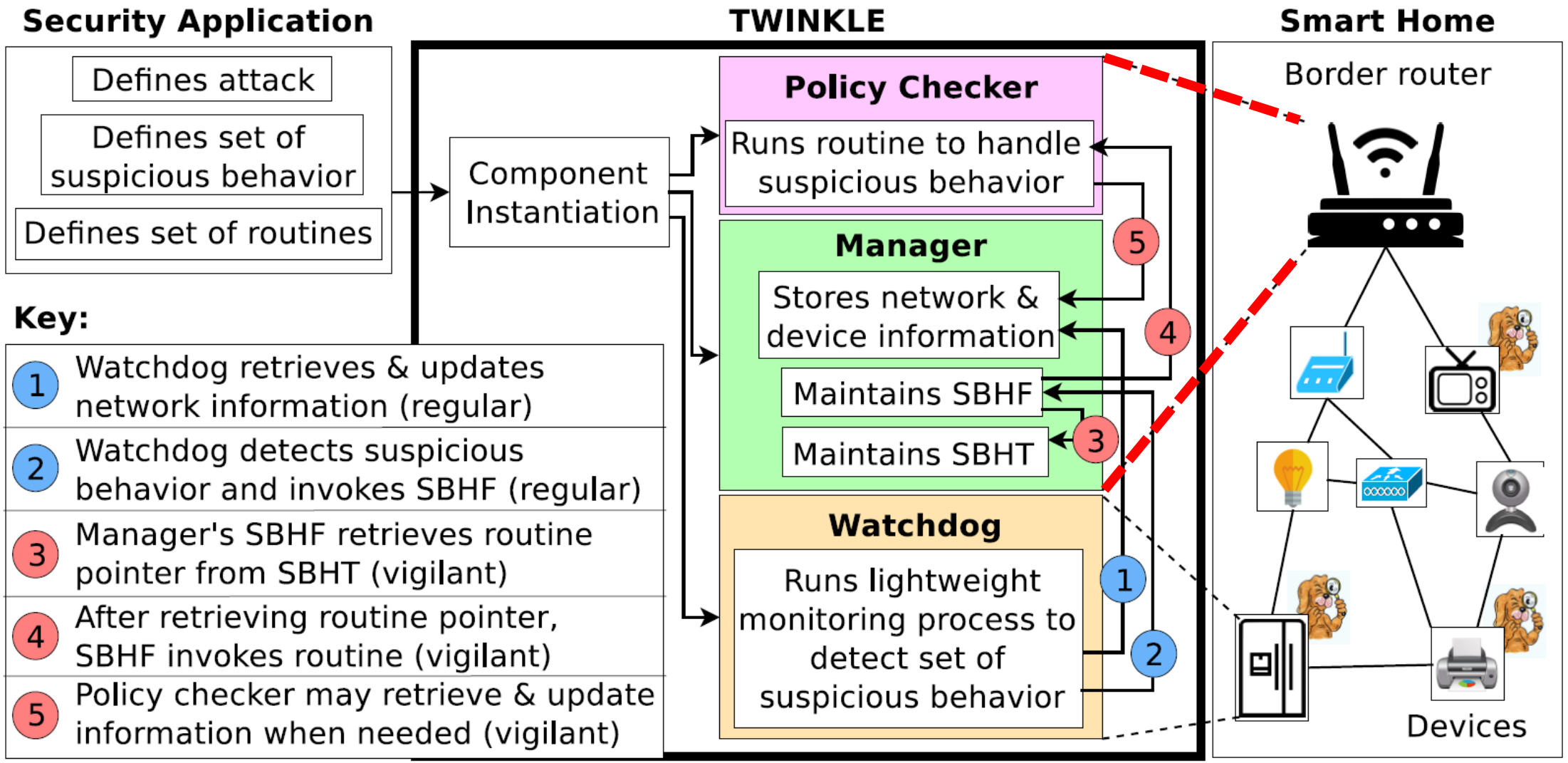
TWINKLE Architecture



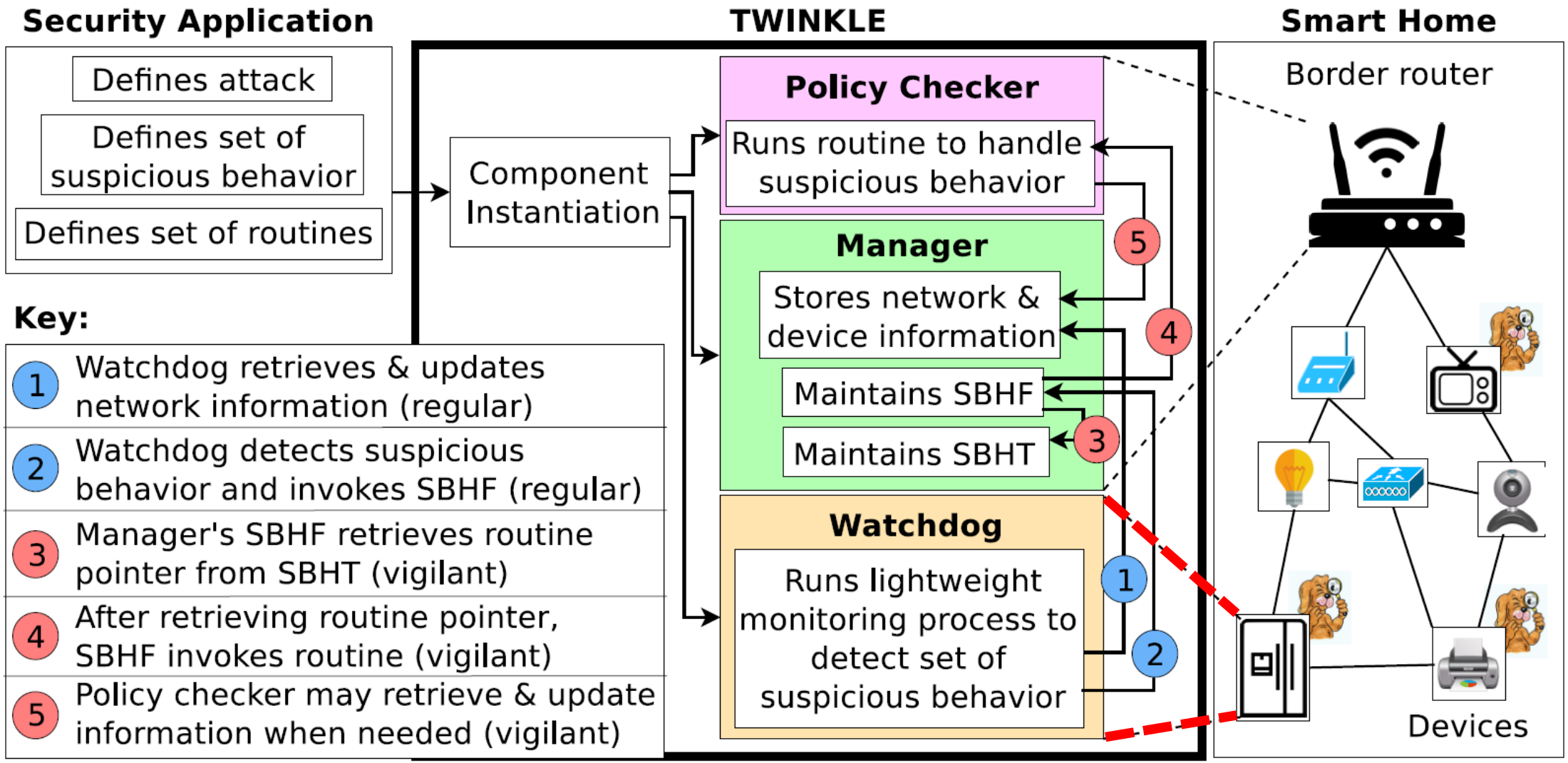
TWINKLE Architecture



TWINKLE Architecture



TWINKLE Architecture



TWINKLE Architecture

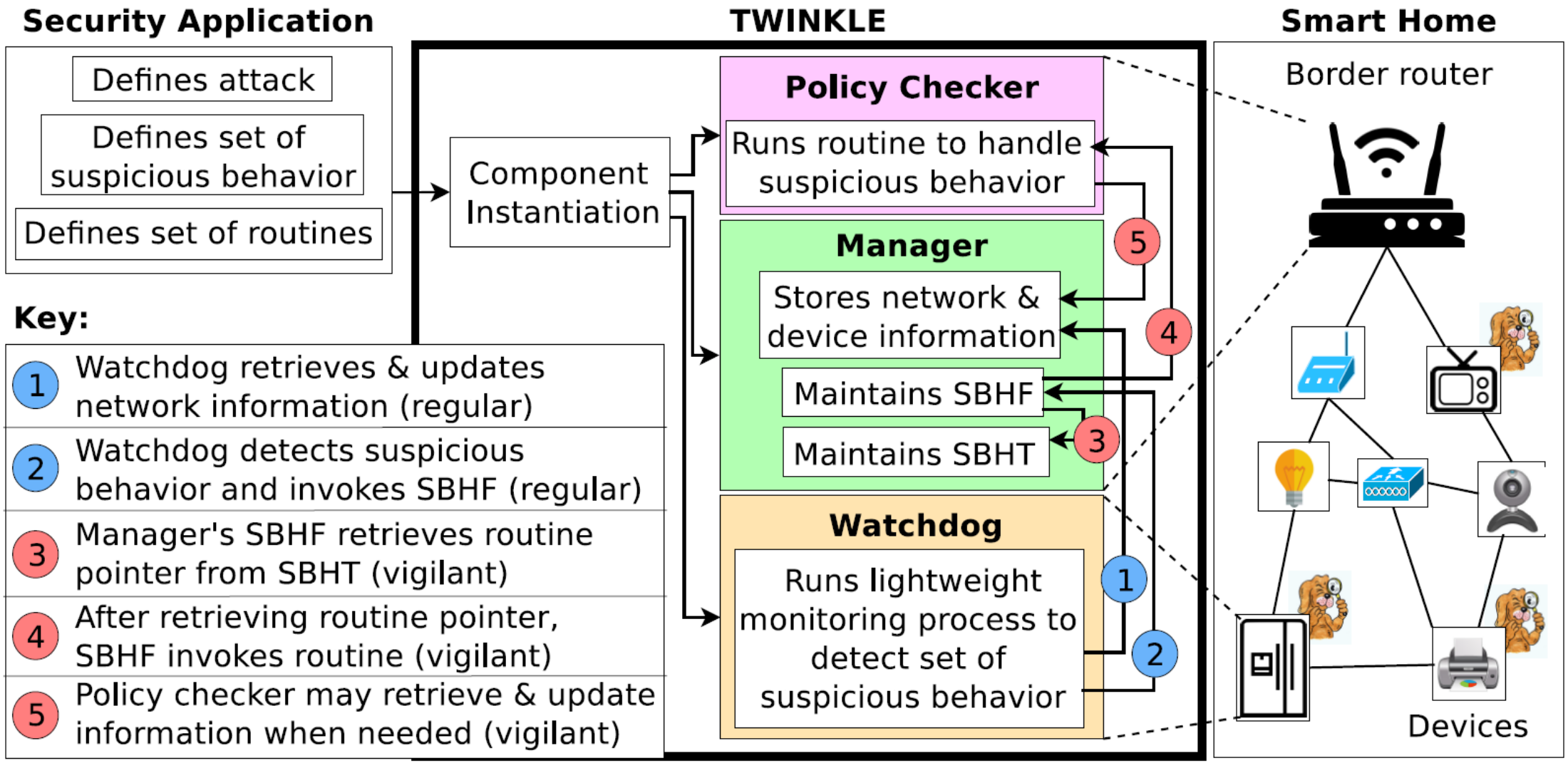
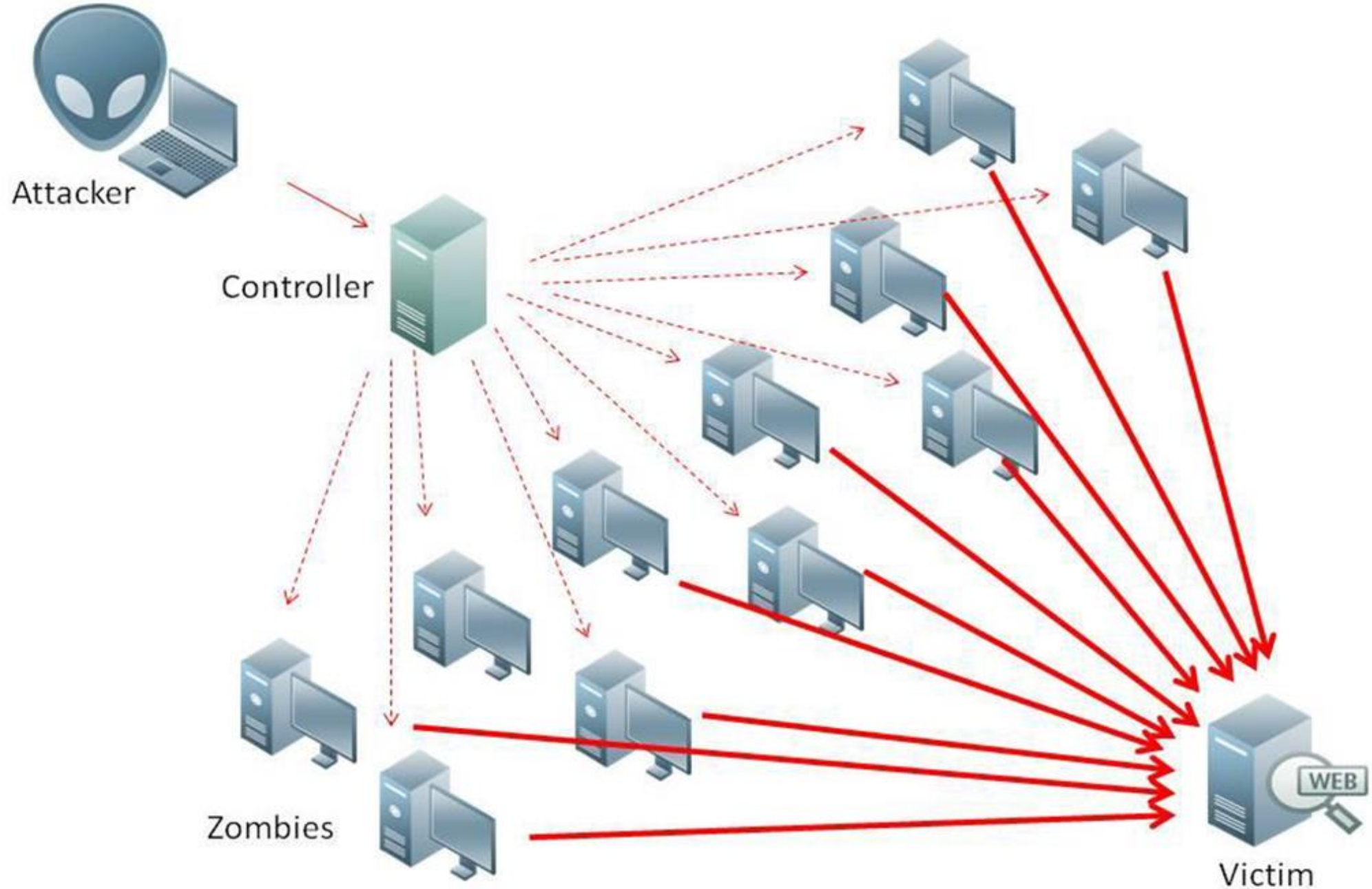
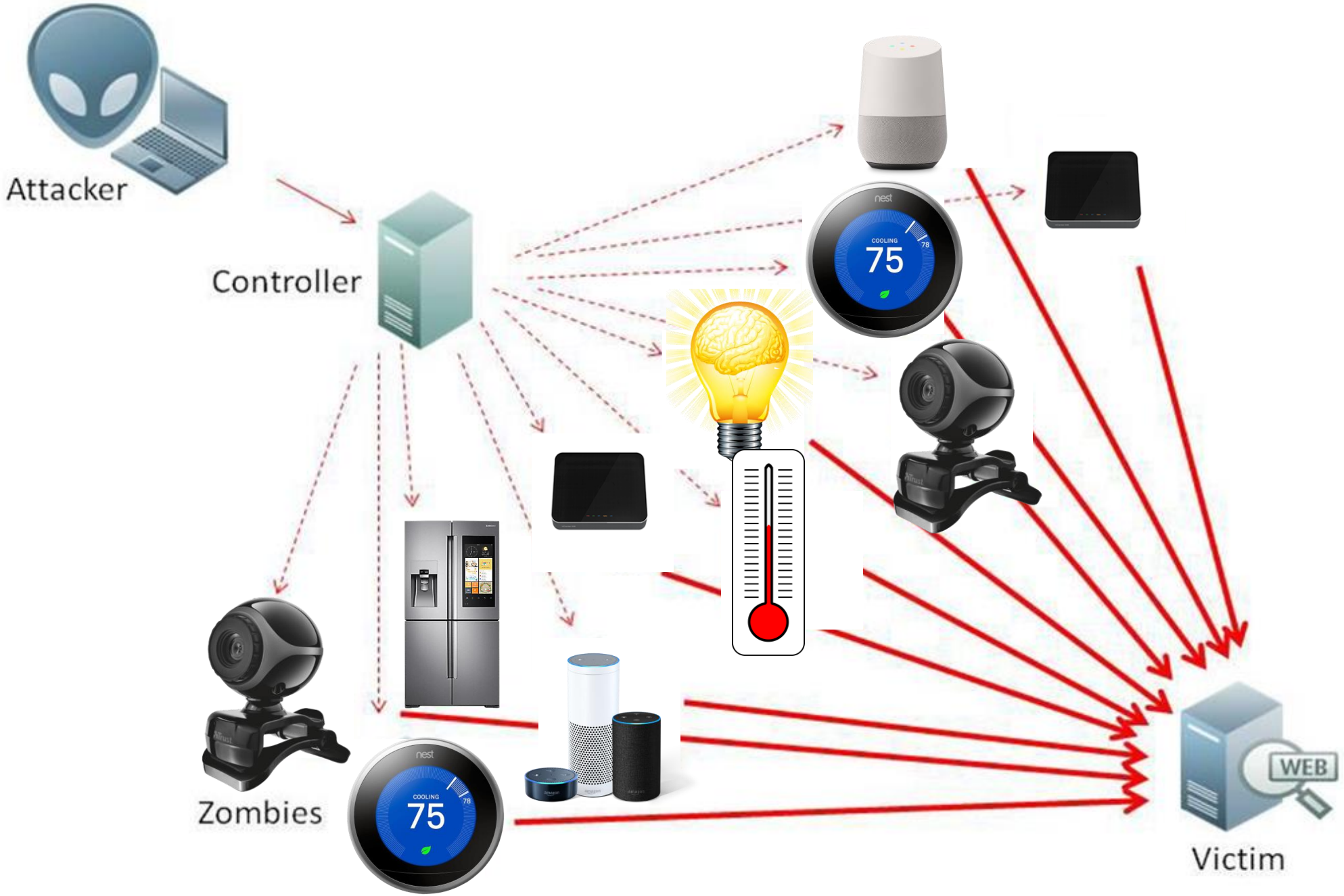


Table of Contents



https://www.realnets.com/our-blog/massive-ddos-attacks-lizardstresser/


Attacker
Controller
Zombies
Victim
WEB
nest
COOLING
75
78

Prior Art: D-WARD Against DDoS Attacks

- Source-end solution – deployed at border router
- Classifies each aggregated flow (agflow) as good, suspicious, or attack
  - agflow – each connection from devices in the network to a specific destination outside the network
- If rate limit is followed for certain period of time, rate limit increases linearly and is eventually removed

Mirkovic, Jelena, and Peter Reiher. "D-WARD: A Source-End Defense Against Flooding Denial-of-Service Attacks." *IEEE Transactions on Dependable and Secure Computing* 2.3 (2005): 216-232.

Prior Art: D-WARD's Shortcomings

- D-WARD targeted towards traditional end-hosts
- Could hurt benign devices if their connections are labeled as transient connections
- While a traditional benign end-host can recover from the accidental loss of their packets, in a IoT environment such as a smart home, a benign device could instead suffer significantly from such a loss

D-WARD+: A Two-Mode Approach Against DDoS Attacks

Essential difference between D-WARD and D-WARD+: D-WARD+ does not drop traffic of transient connections, but instead leverages the *fast retransmit* mechanism of TCP congestion control

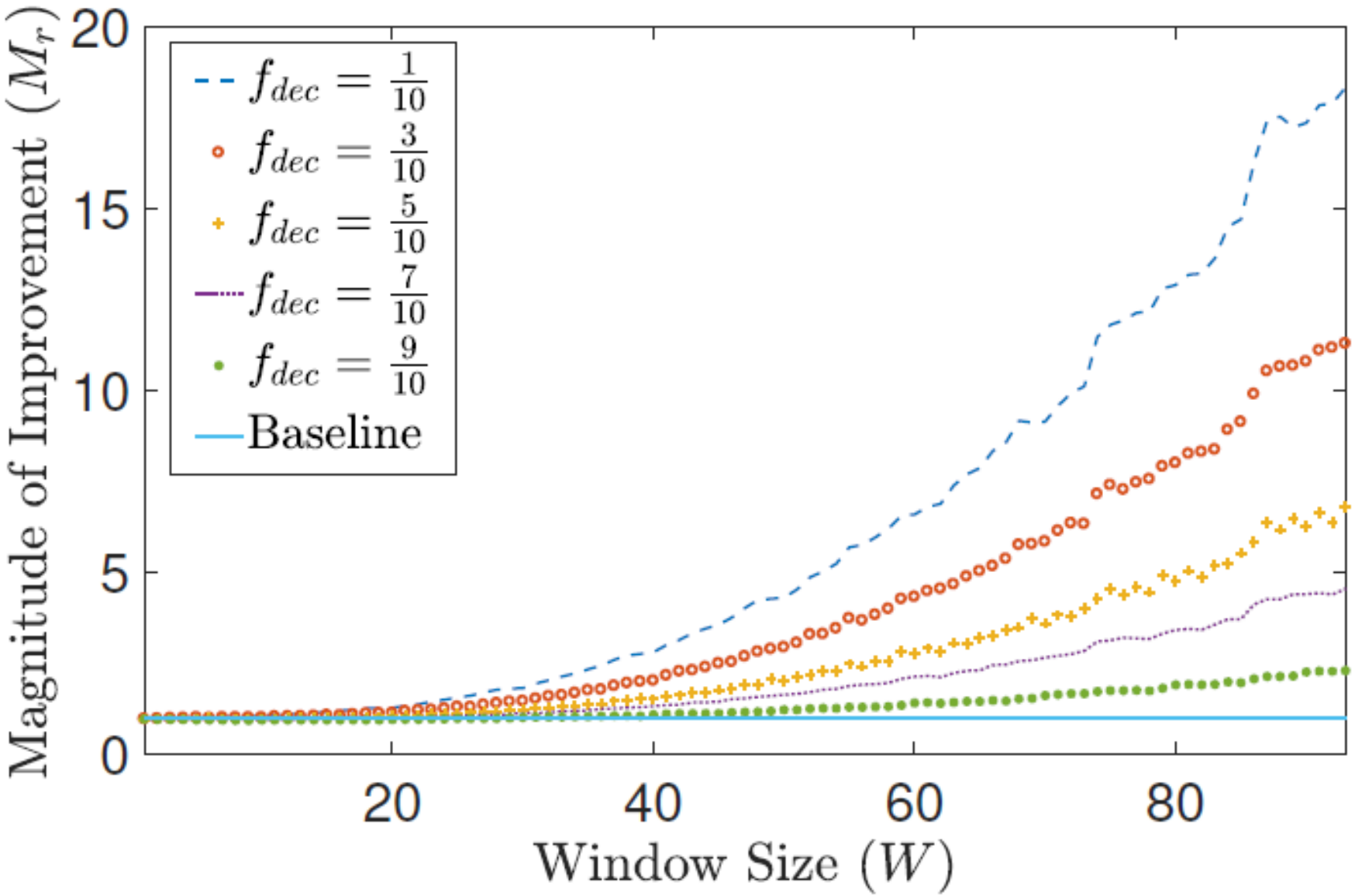
D-WARD+: A Two-Mode Approach Against DDoS Attacks

1. In regular mode:
   - Watchdog passively monitors network
   - Once attack agflow is detected, watchdog notifies manager
   - Manager's SBHF is invoked and system enters vigilant mode
2. In vigilant mode:
   - Manager's SBHF invokes agflow monitoring routine inside policy checker
   - Agflow monitoring routine monitors each transient connection of attack agflow
   - Alert device to cut sending rate in half if rate-limit is surpassed by sending it three duplicate ACK packets (*signal*)
   - If device ignores signals, classify connection as bad and drop traffic
   - Return to regular mode once all connections are below the rate-limit for a certain amount of time and receivers are not congested (determined by flow control)

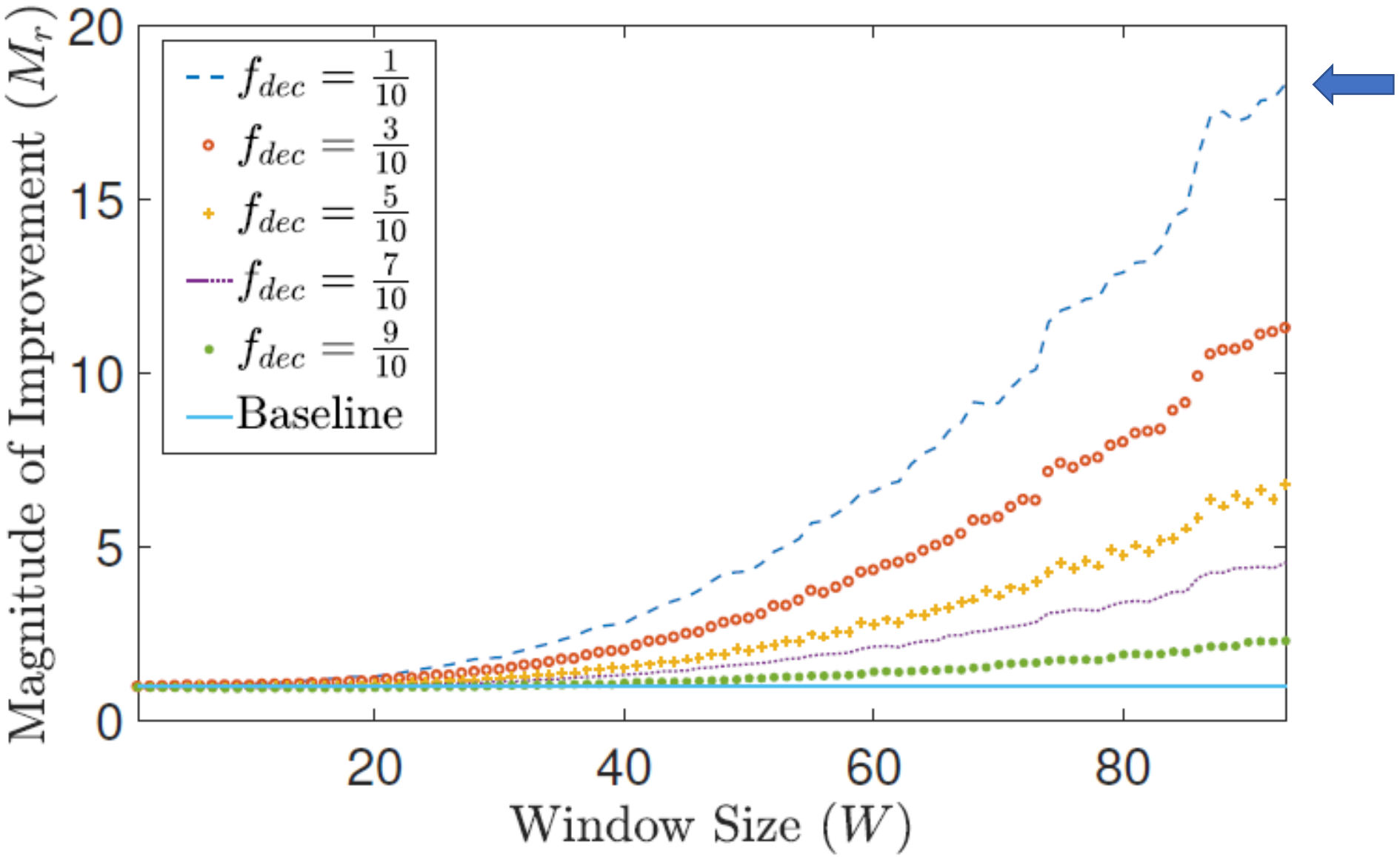
Evaluation Setup

- Implemented D-WARD+ and D-WARD in Java
- 2015 Dell XPS: 2.2 GHz Intel Core i5 processor, 8GB of RAM
- Constructed a Bluetooth Personal Area Network (PAN)
- Client device transfers 2.5 MB of data to the server through a router on which D-WARD+ and D-WARD are implemented
- Client device performs simple and smart TCP flooding attacks
- Client and server utilized TCP New Reno for congestion control

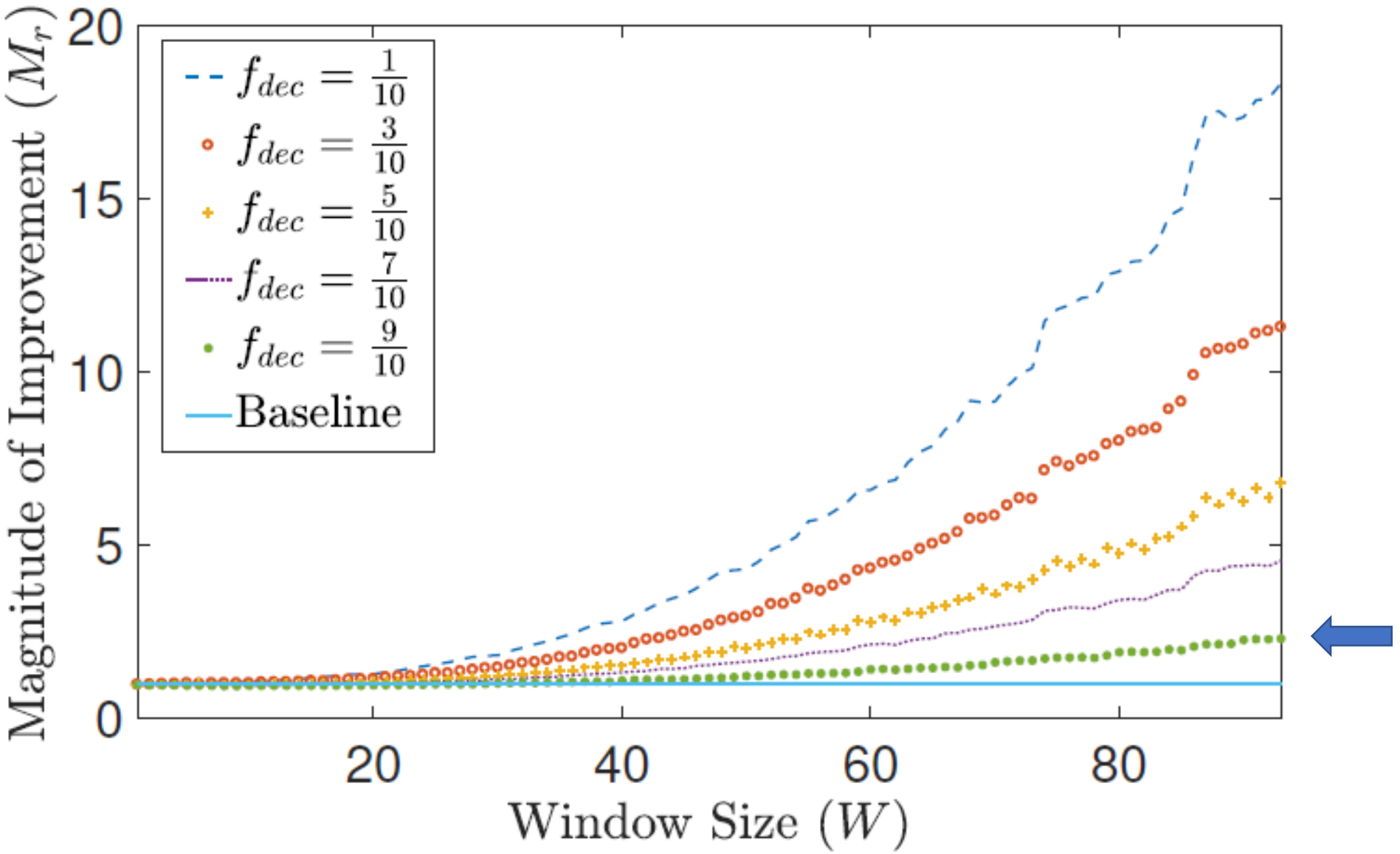
Improvement in Retransmissions



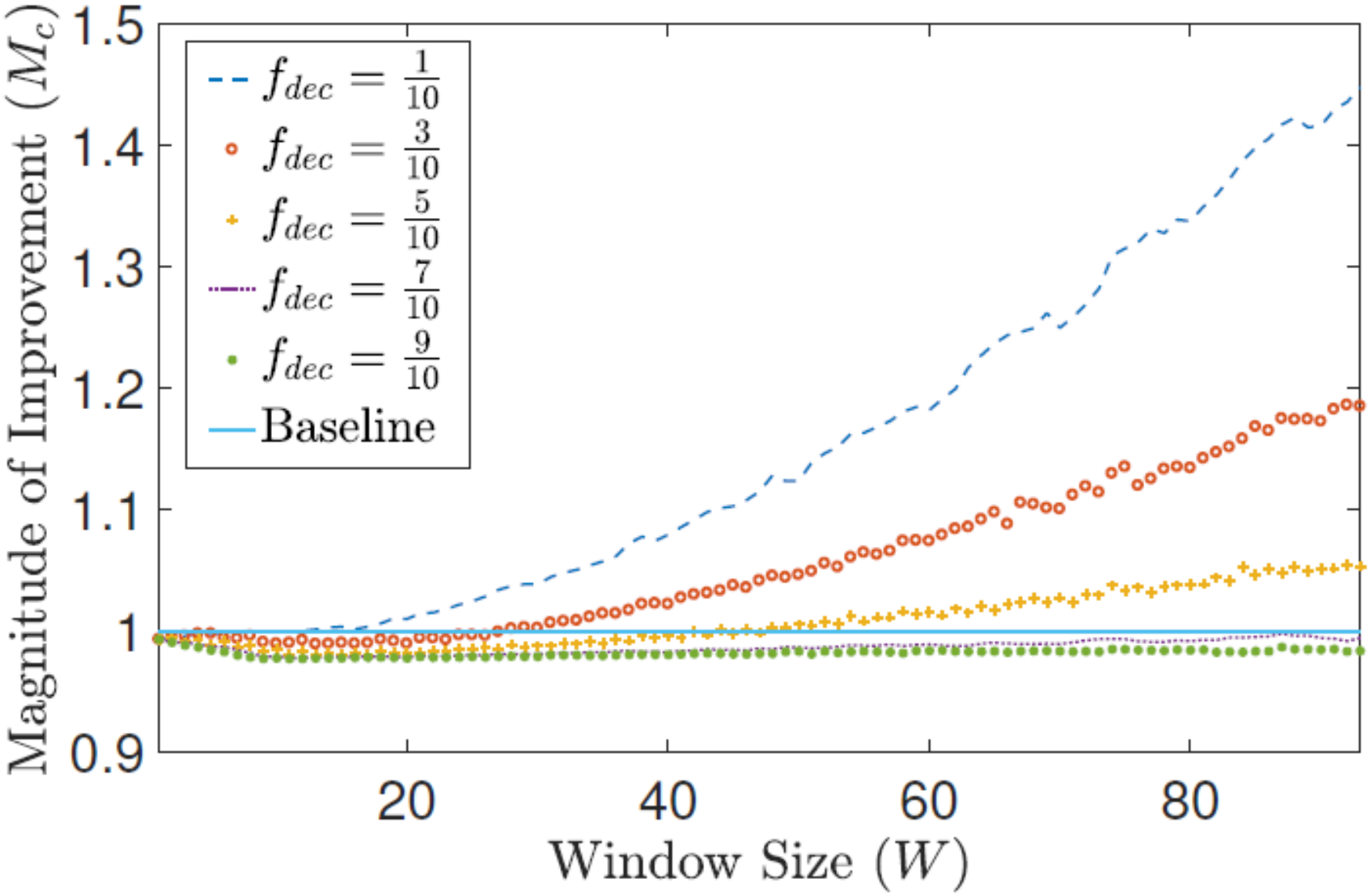
Improvement in Retransmissions



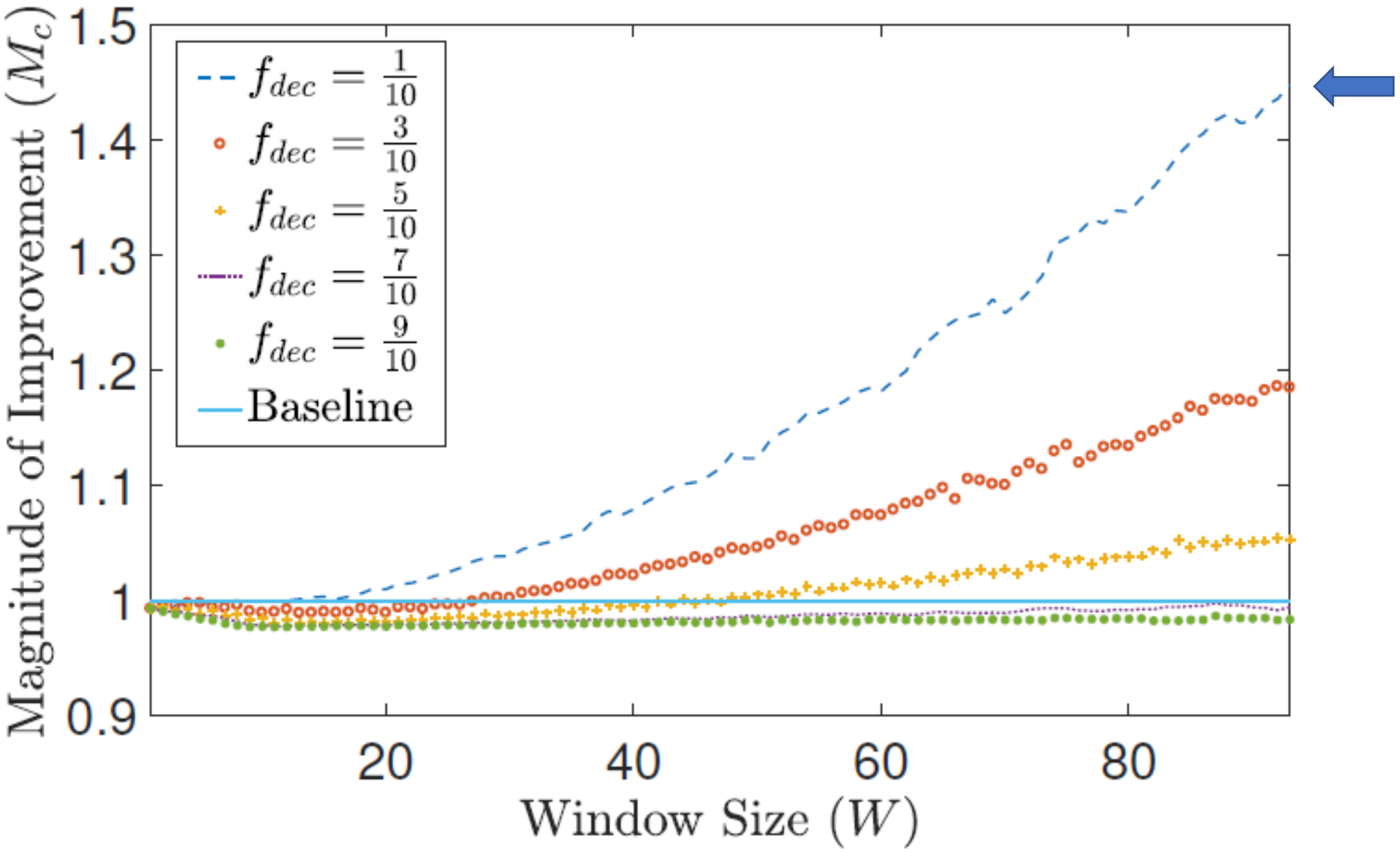
Improvement in Retransmissions



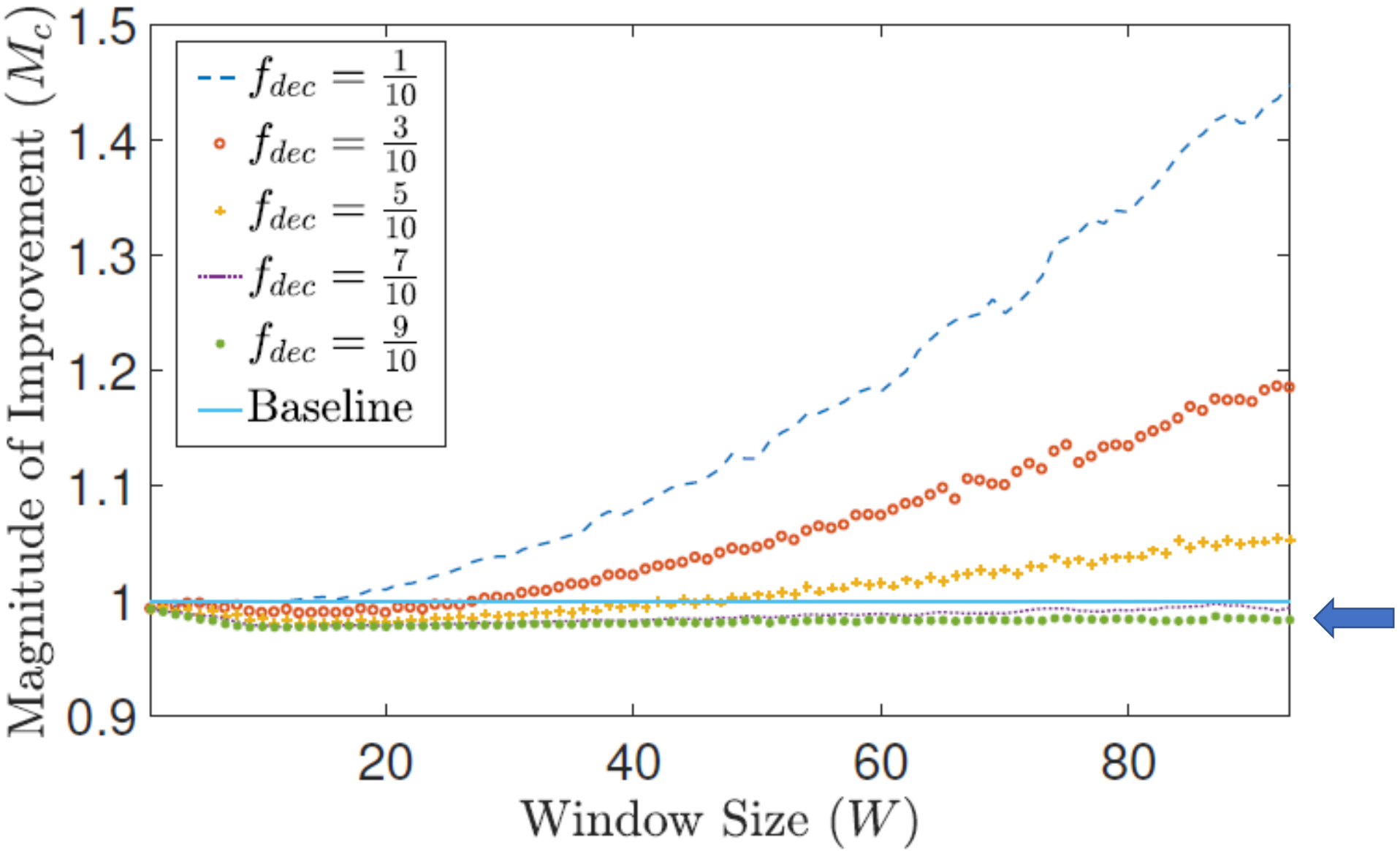
Improvement in Connection Duration



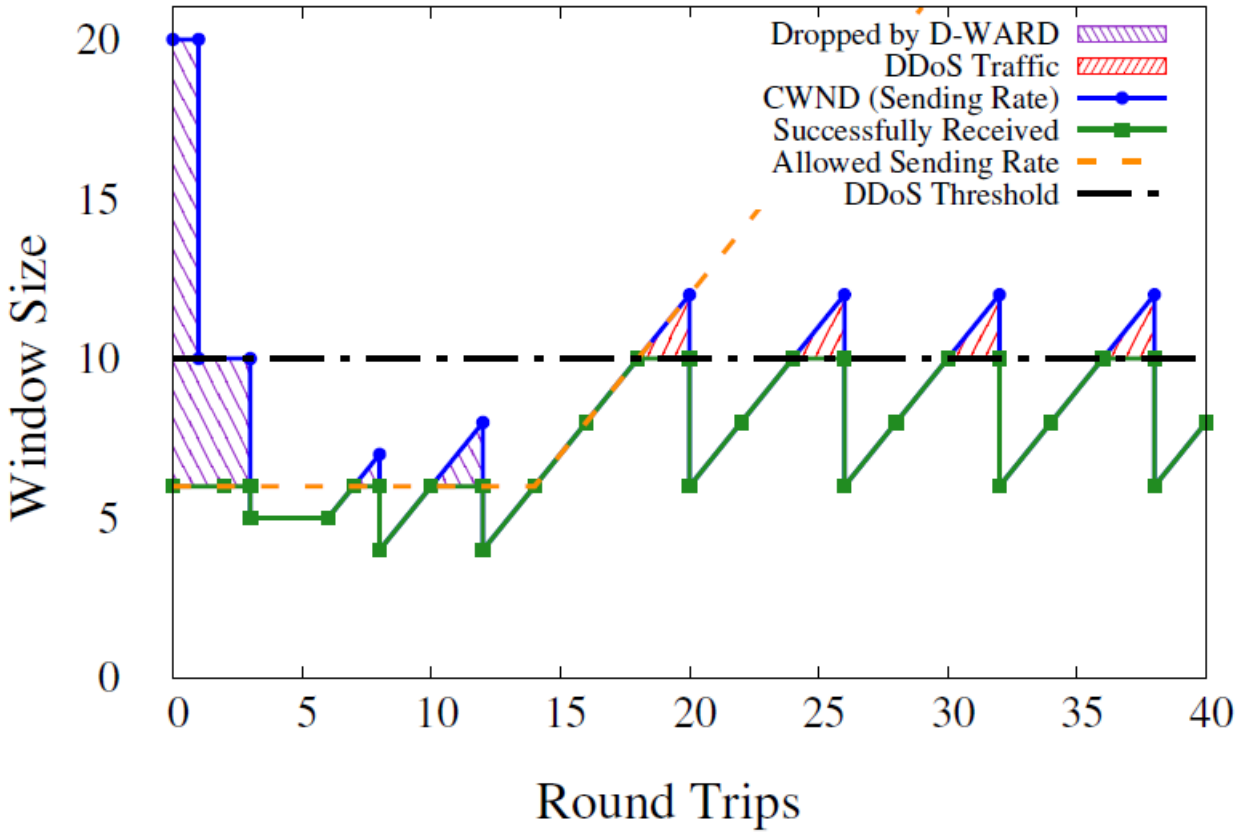
Improvement in Connection Duration



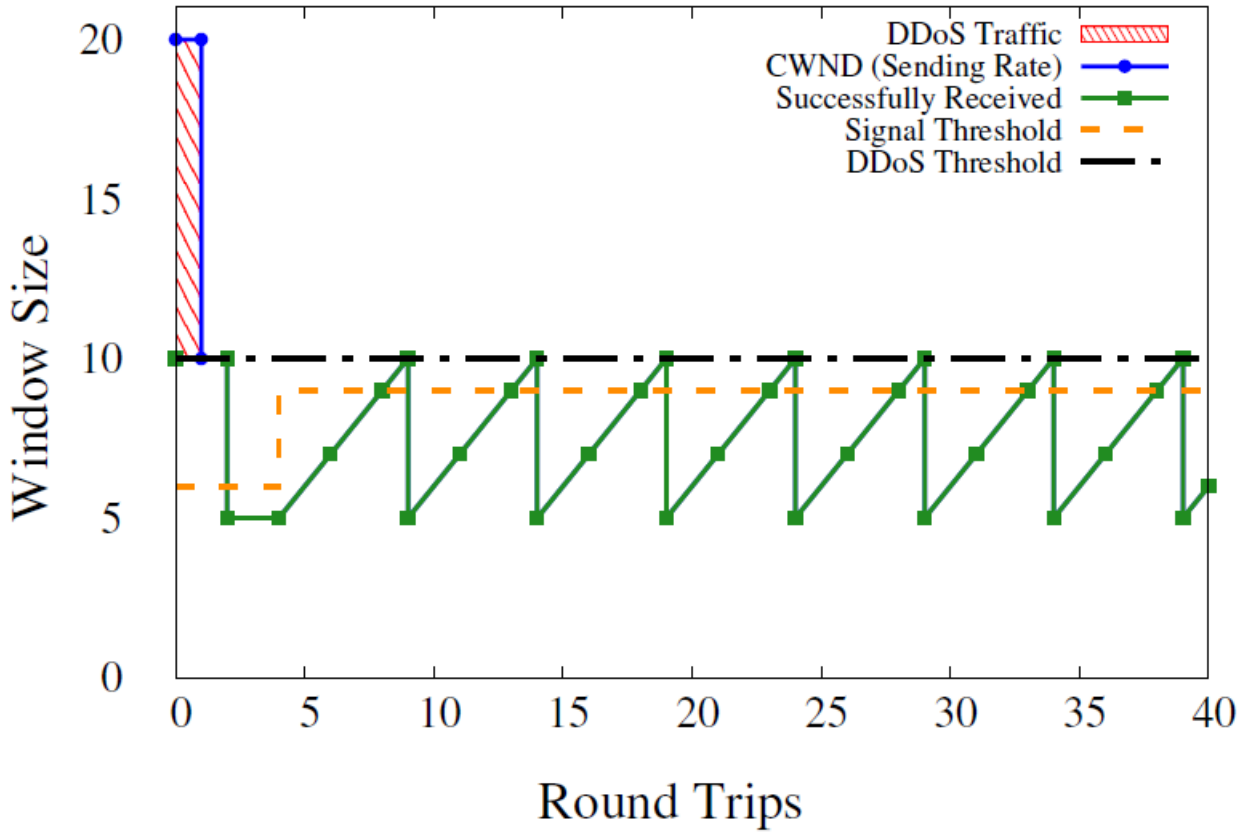
Improvement in Connection Duration



Effect on “Smart” Attacker

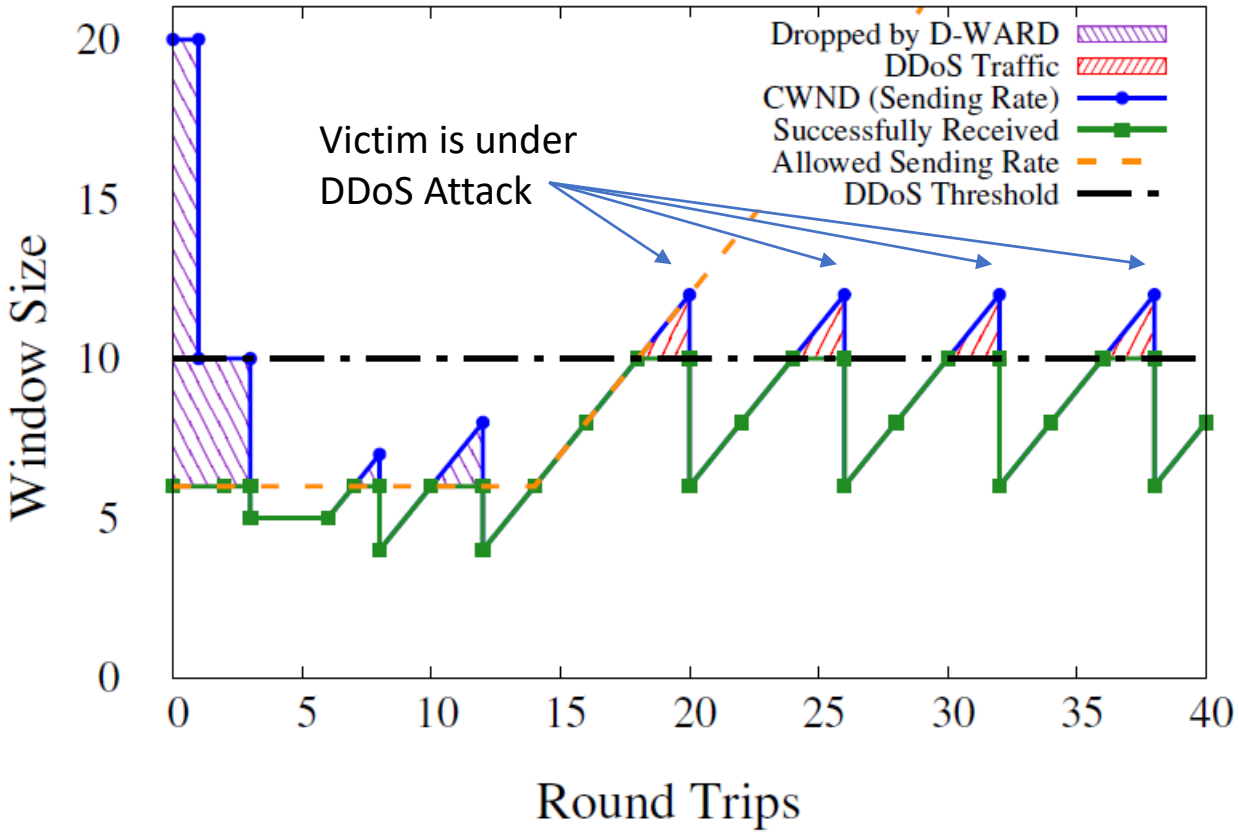


(a) D-WARD

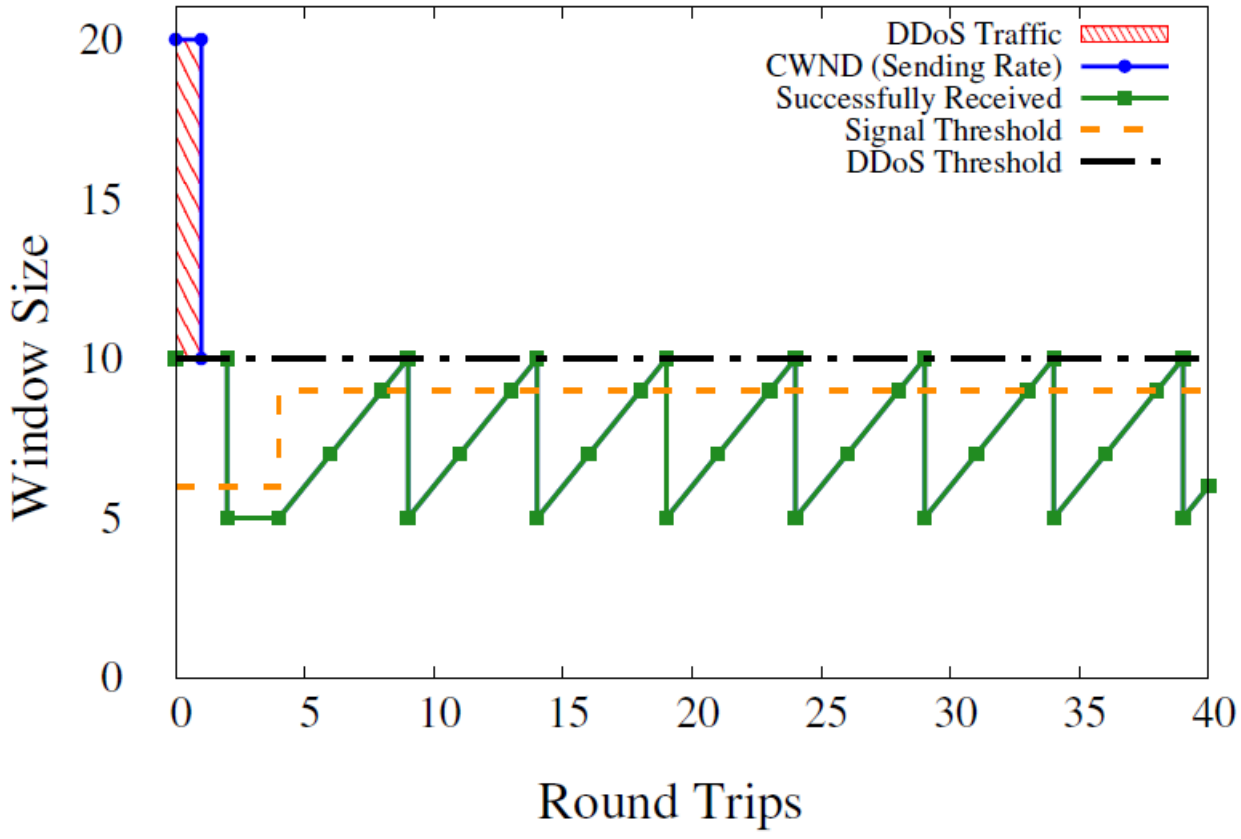


(b) D-WARD+

Effect on “Smart” Attacker



(a) D-WARD



(b) D-WARD+

D-WARD+ vs. D-WARD

- Retransmission of packets: by minimizing the dropping of packets from benign devices, D-WARD+ reduces amount of retransmissions of benign devices
  - Thus reducing network overhead and battery consumption
- Connection duration: in most cases, by not dropping packets, D-WARD+ reduces connection duration
  - Thus reducing data transfer time

Table of Contents

Sinkhole Attack

- A compromised device announces a short path toward a destination node to attract traffic from other nodes to the destination, therefore intercepting or dropping the traffic and creating a sinkhole
- Sinkhole attack via RPL (Routing Protocol over Low Powered and Lossy Networks) can happen when a device sends an advertisement message to its neighbors to lie that the device has a low rank
- RPL's self-healing and repair mechanisms are not resilient against sinkhole attacks

Prior Art: SVELTE Against the Sinkhole Attack in 6LoWPAN

- Two main modules running on the border router of a 6LoWPAN network:
  1. A mapper which builds a routing map of the network
  2. Intrusion detection module which checks for the rank inconsistencies
- One module running on the devices:
  - Receives and responds to probing messages from border router

Raza, Shahid, Linus Wallgren, and Thiemo Voigt. "SVELTE: Real-time intrusion detection in the Internet of Things." *Ad Hoc Networks* 11.8 (2013): 2661-2674.

Prior Art: SVELTE's Shortcomings

- SVELTE's probing mechanism can increase network overhead, device battery consumption, and latency of detection
- Every probe from the border router will increase network overhead
- Every response from a device will increase network overhead and consume device's battery
- Worst of all, SVELTE has a dilemma in choosing the probing interval:
  - A short interval will lead to a low latency in detecting sinkhole attacks, but a large overhead due to frequent probing and responding
  - A long interval will result in a low overhead, but a high latency in detecting sinkhole attacks

SVELTE+: A Two-Mode Approach Against 6LowPAN Sinkhole Attacks

Essential difference between SVELTE+ and SVELTE: Border router will NOT probe the entire network periodically, but **on demand**

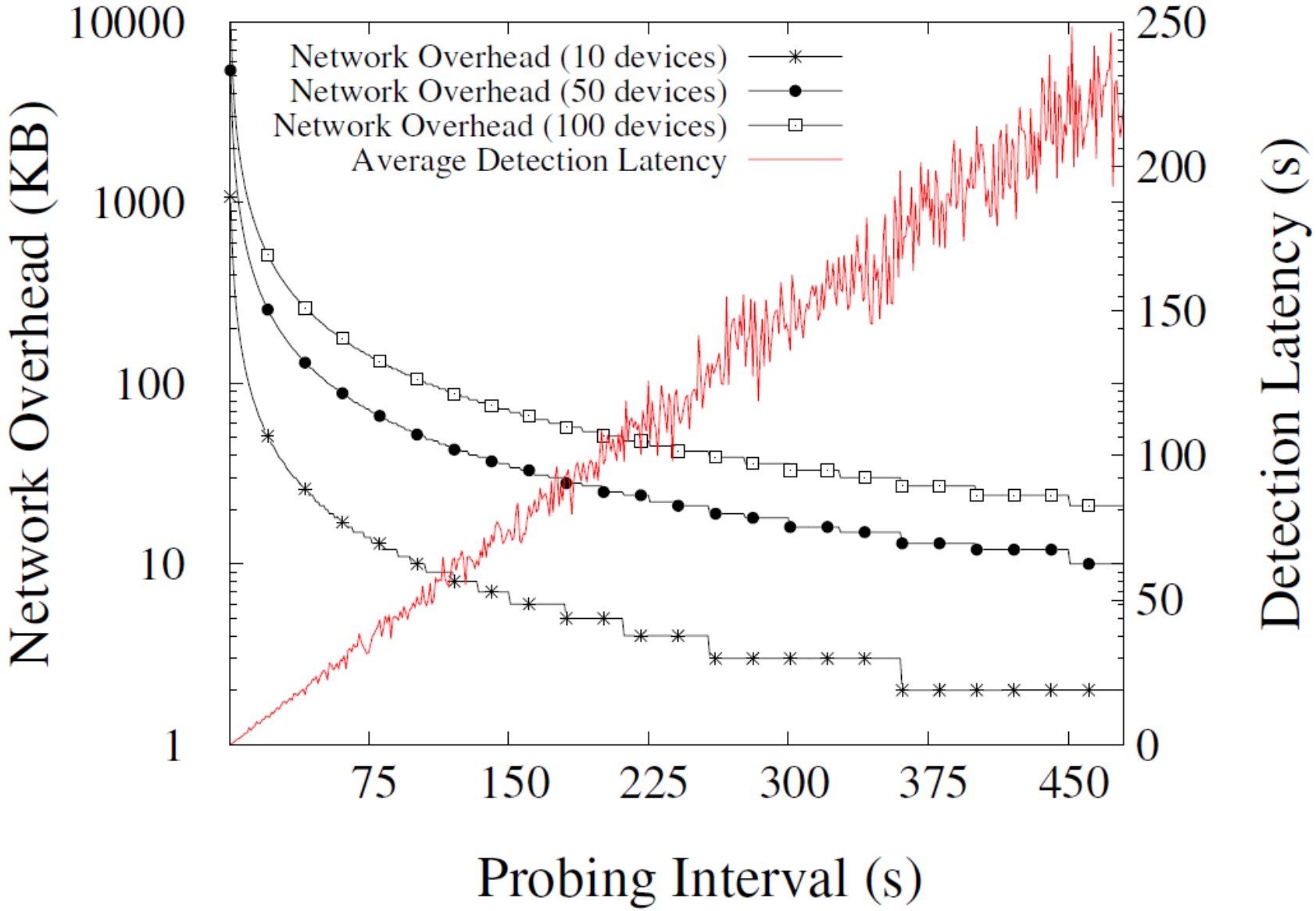
SVELTE+: A Two-Mode Approach Against 6LowPAN Sinkhole Attacks

1. In regular mode:
   - Watchdogs passively monitors network
   - Once a *new* rank advertisement occurs, all watchdogs in range of advertisement will notify manager
   - Manager's SBHF is invoked and system enters vigilant mode
2. In vigilant mode:
   - Manager's SBHF invokes sinkhole detection routine inside policy checker
   - Sinkhole detection routine queries manager for up-to-date routing map
   - Compares rank of watchdogs and suspect to find inconsistencies
     a) If inconsistency NOT found: update routing map and return to regular mode
     b) If inconsistency is detected: Policy checker invokes sinkhole mitigation routine and once routine is complete, return to regular mode

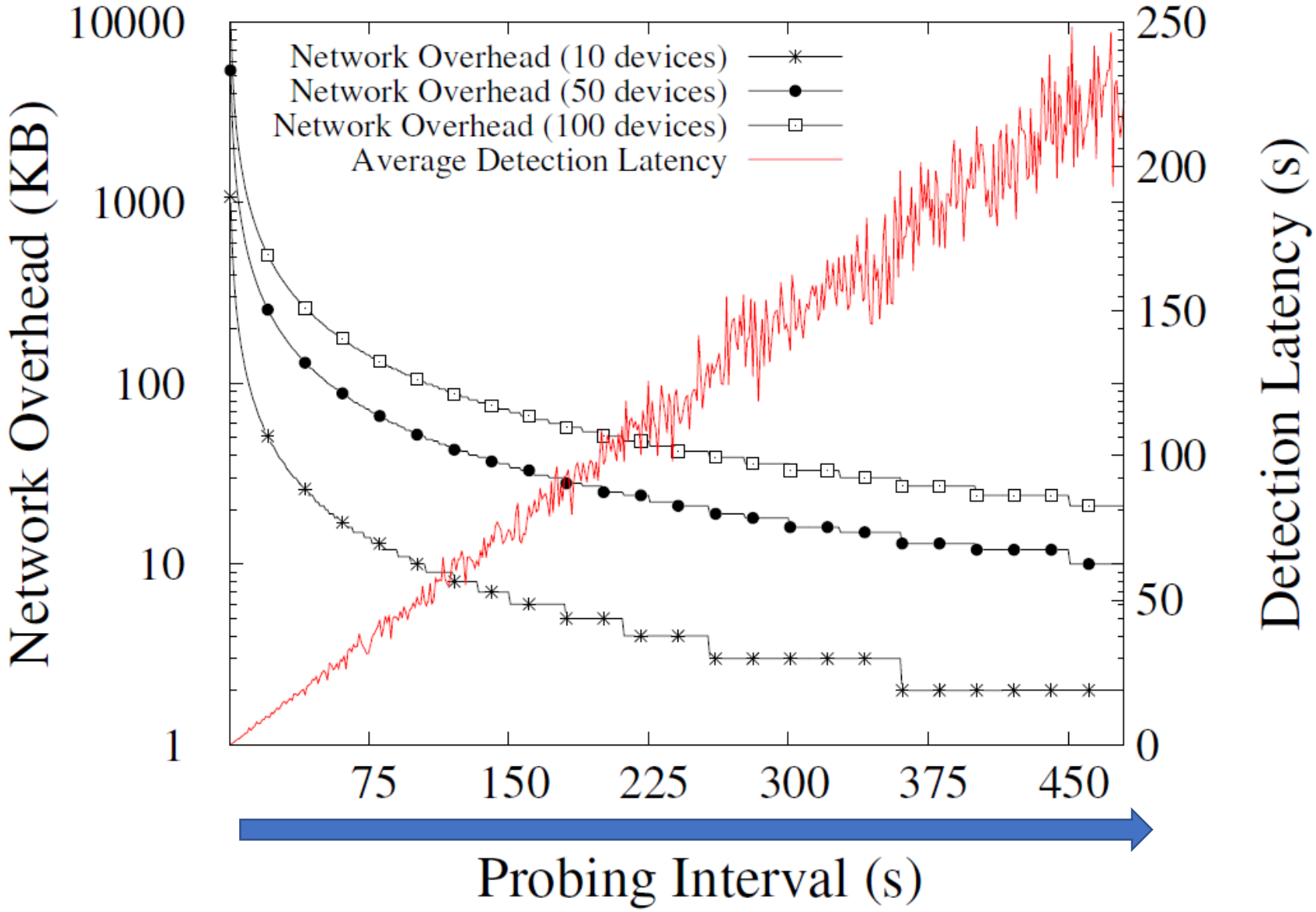
Evaluation Setup

- Implemented SVELTE+ and SVELTE in Java
- 2015 Dell XPS: 2.2 GHz Intel Core i5 processor, 8GB of RAM
- Randomly generated mesh IoT network topologies of varying size
- Simulated RPL and sinkhole attacks through rank manipulation

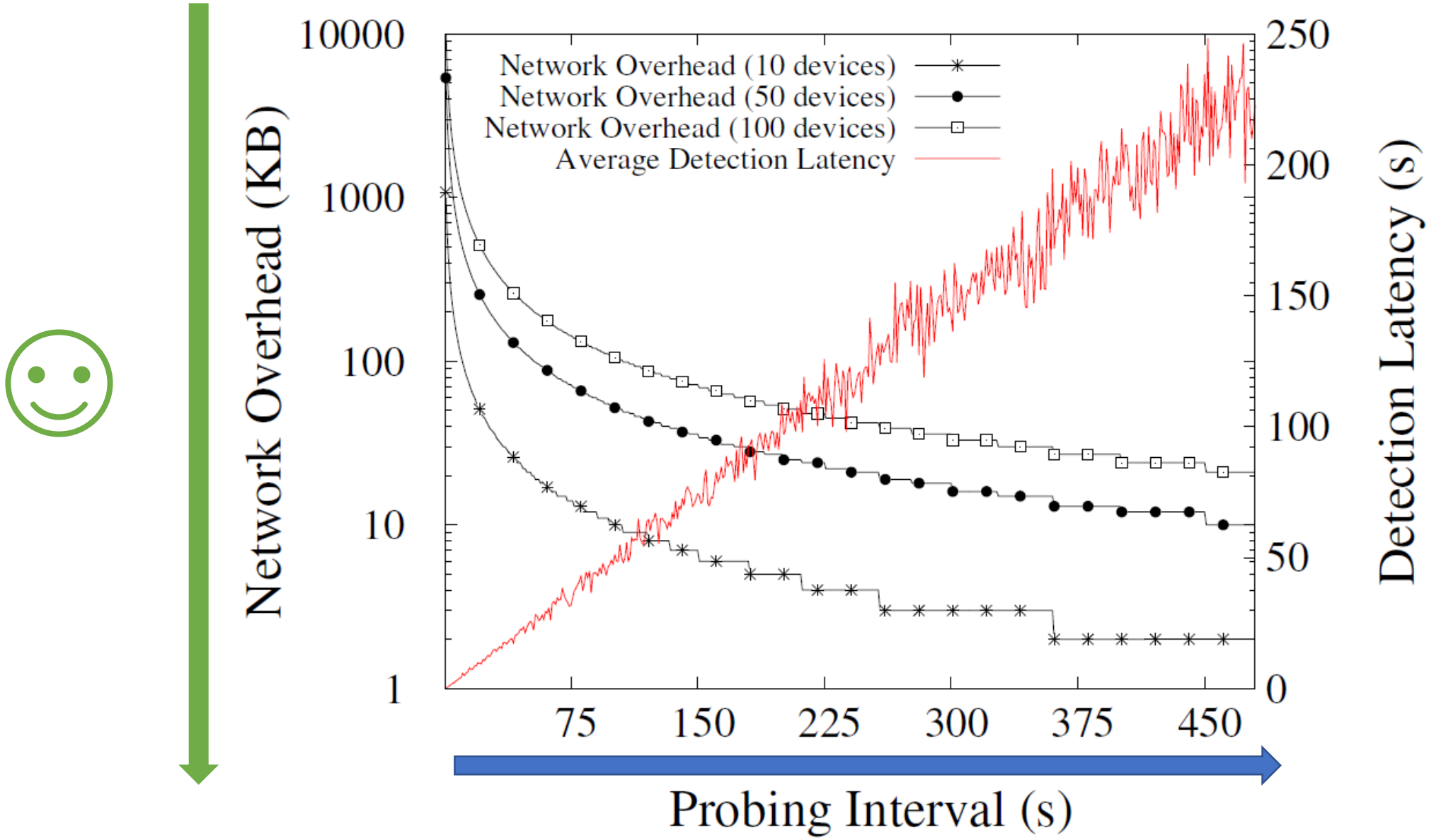
Effect of Probing Interval in SVELTE



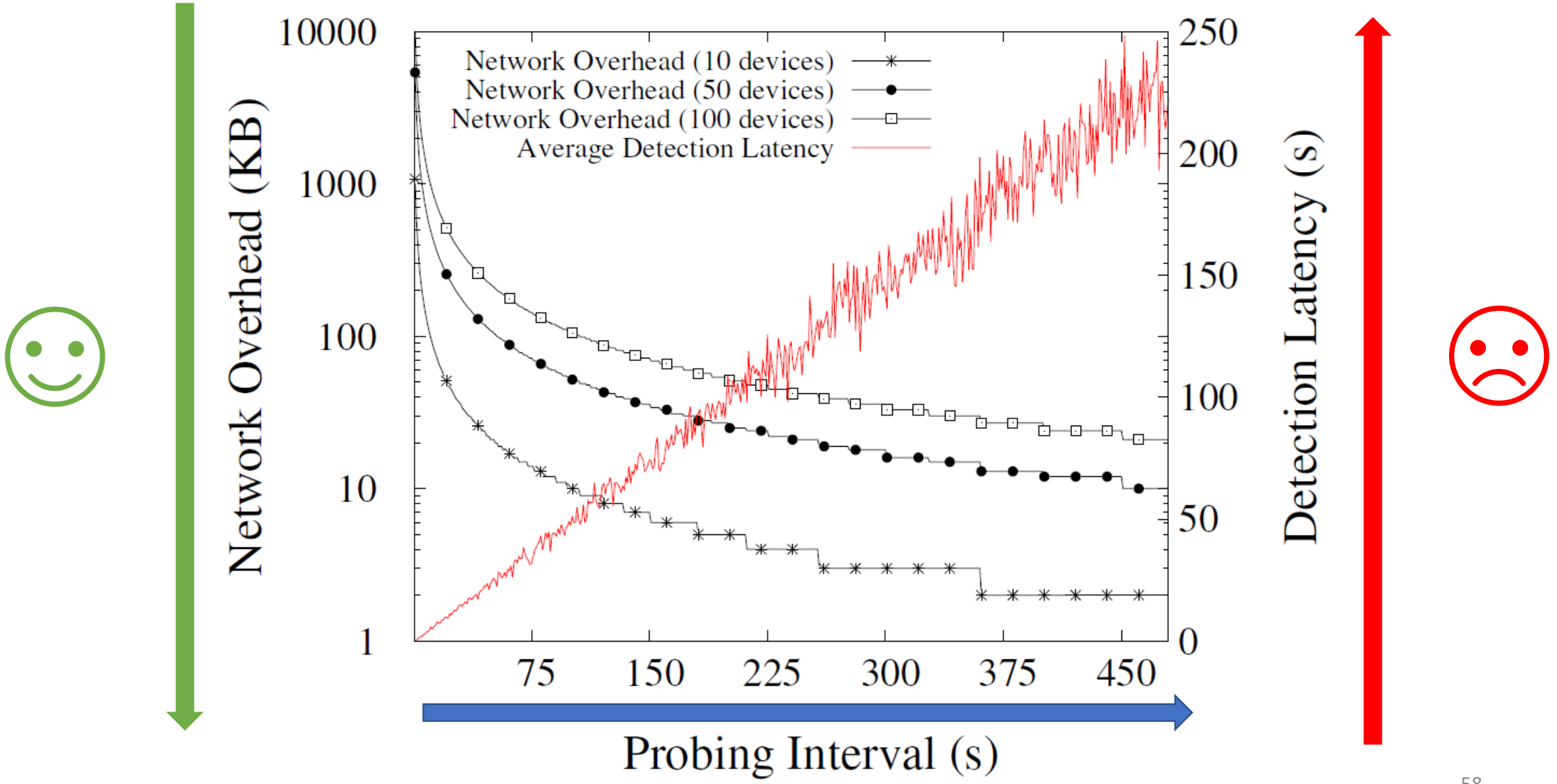
Effect of Probing Interval in SVELTE



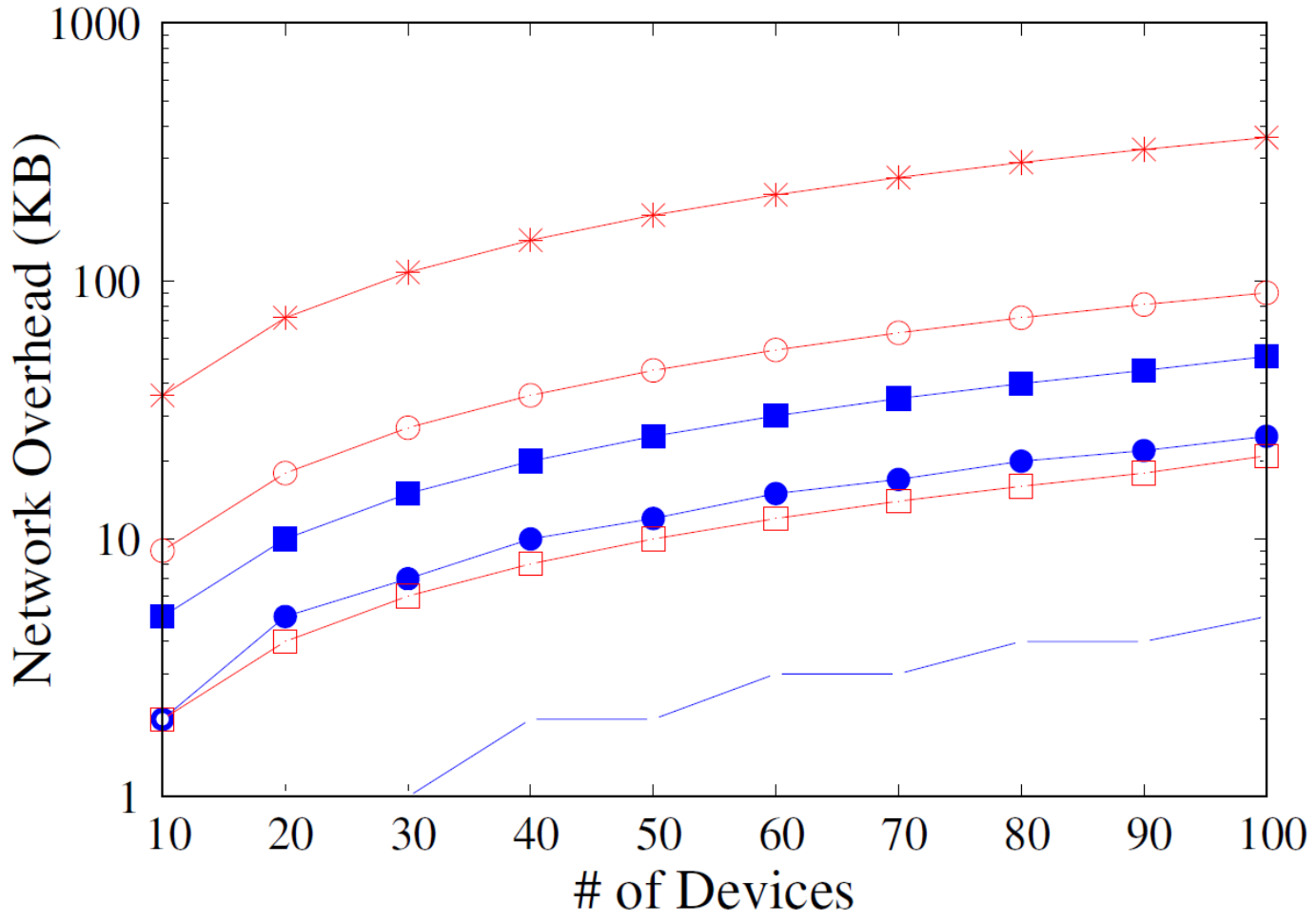
Effect of Probing Interval in SVELTE



Effect of Probing Interval in SVELTE



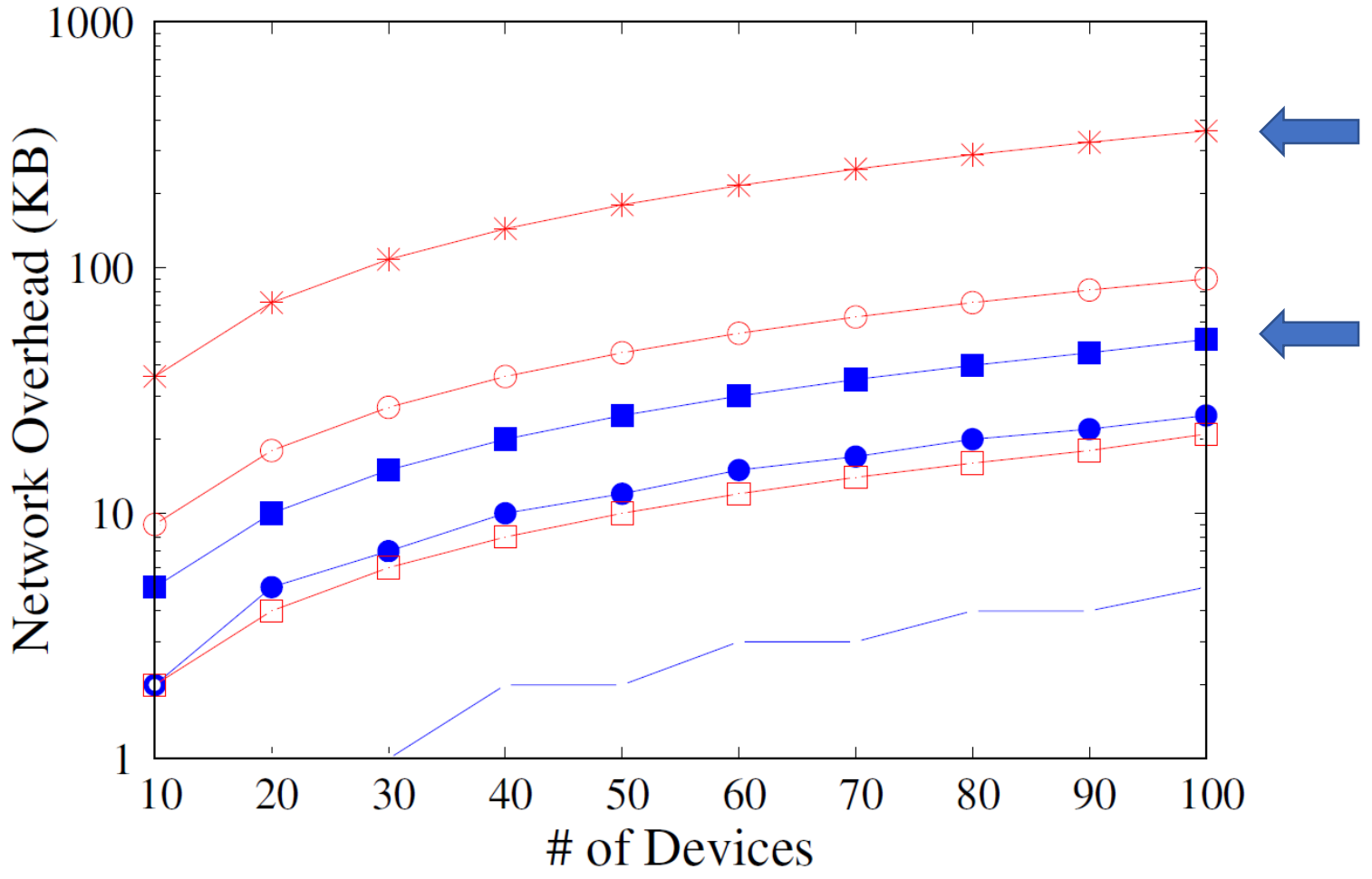
Difference in Network Overhead



PI: Probing Interval

NRA/D: New Rank Advertisements per Device

Difference in Network Overhead

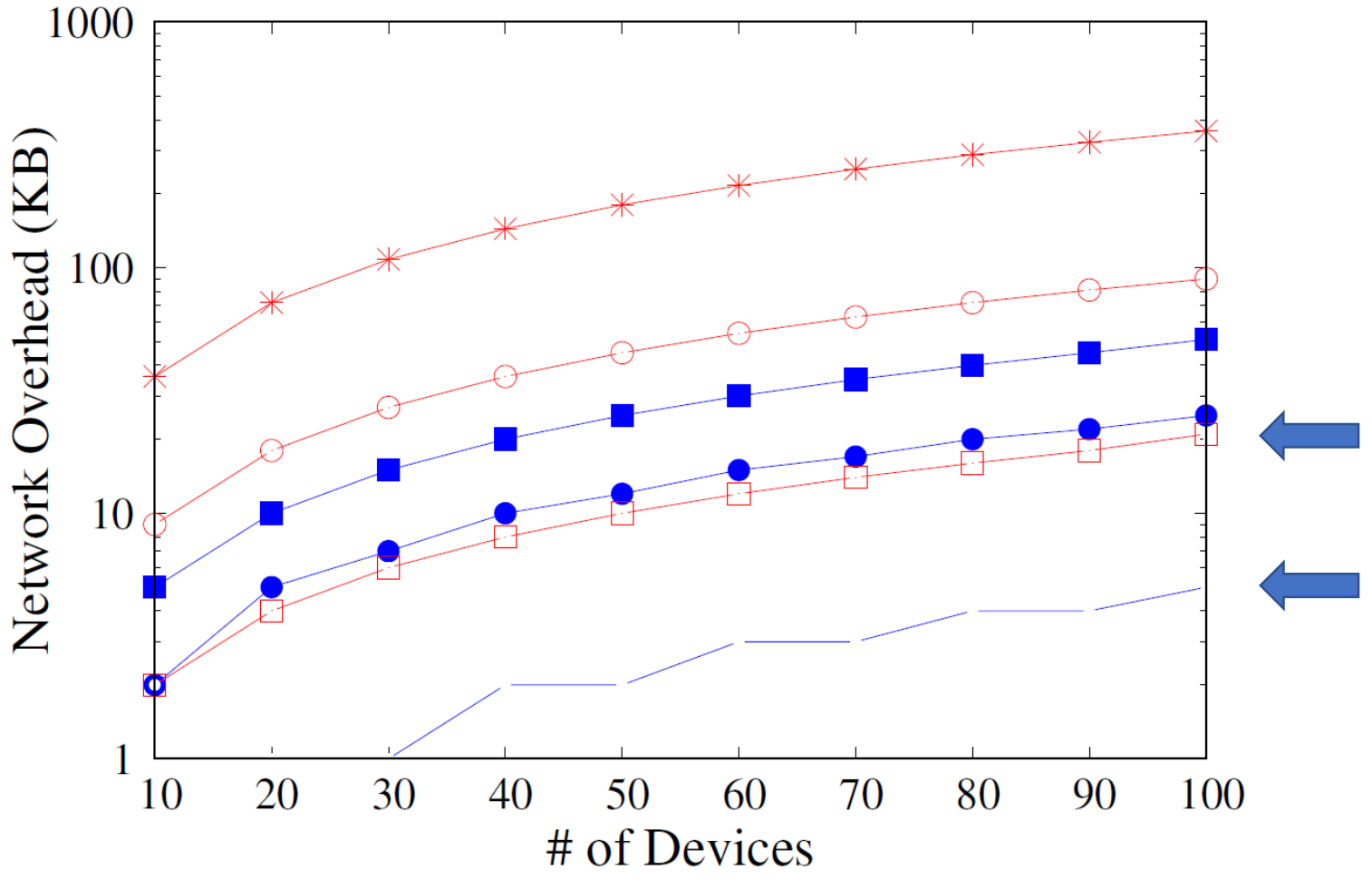


SVELTE+ (100 NRAs)
SVELTE+ (500 NRAs)
SVELTE+ (1000 NRAs)
SVELTE (30s PI)
SVELTE (120s PI)
SVELTE (480s PI)

PI: Probing Interval

NRA/D: New Rank Advertisements per Device

Difference in Network Overhead



SVELTE+ (100 NRAs)
SVELTE+ (500 NRAs)
SVELTE+ (1000 NRAs)
SVELTE (30s PI)
SVELTE (120s PI)
SVELTE (480s PI)

PI: Probing Interval

NRA/D: New Rank Advertisements per Device

SVELTE+ vs. SVELTE

- Detection latency: negligible in SVELTE+
  - SVELTE+ does not wait for probing interval to check network
- Network overhead and battery consumption: SVELTE+ may incur more overhead in the beginning as nodes join the network, but as the network stabilizes, the amount of times SVELTE+ switches to vigilant mode will be low

Table of Contents

Feasibility and Limitations of TWINKLE

- Installing manager and policy checker on border router may not be feasible in certain environments
- Installing watchdog on devices may also not be feasible in certain environments
  - Installing on legacy devices?
  - Installing on low-powered devices?
  - Some environments may need to deploy devices whose sole purpose is to run Watchdog code

Table of Contents

Conclusion

- Contributions:
  - TWINKLE, a two-mode security framework that supports individual security applications that can handle specific attacks in a smart home environment
  - Two-mode methodology: only run enough security features to detect suspicious behavior and add more security features **on demand** if needed
  - Transformed two existing security solutions and made them more resource-efficient
- Future Work:
  - Implementing TWINKLE on a real IoT network
  - Extending TWINKLE to include more than two modes

Thank you!

Appendix

Introduction


Hackers ca
spy on you
Samuel Gib
Security re
microphon
ZDNet
Cisco: Most IoT projects are failing due to lack of experience and security
By Corinne Reichert | November 13, 2017 -- 06:33 GMT (22:33 PST) | Topic: Mobility
WI
ANDY GREENBERG SECURITY 10.20.1
THE RE
BOTNE
ALREADY
INFECTED A
MILLION
NETWORKS
BBC
NEWS
Technology
Connected toys have 'worrying' security issues
14 November 2017 | Technolog | 50
nadows in the Past Month
October 20, 2017 09:30 AM 1
orm?
Why KRACK could hit your smart home's Wi-Fi the hardest
Smart devices and appliances, everything from your refrigerator to your television, are ideal targets for hackers thanks to the KRACK exploit.
BY ALFRED NG / OCTOBER 16, 2017 12:38 PM PDT
Hello Barbie
back. Photograp

c|net

IoT attacks are getting
liste

threat post

IEEE SPECTRUM

Opinion | Telecom | Security

20 Jun 2018 | 19:00 GMT

6 Reasons Why IoT Security Is Terrible

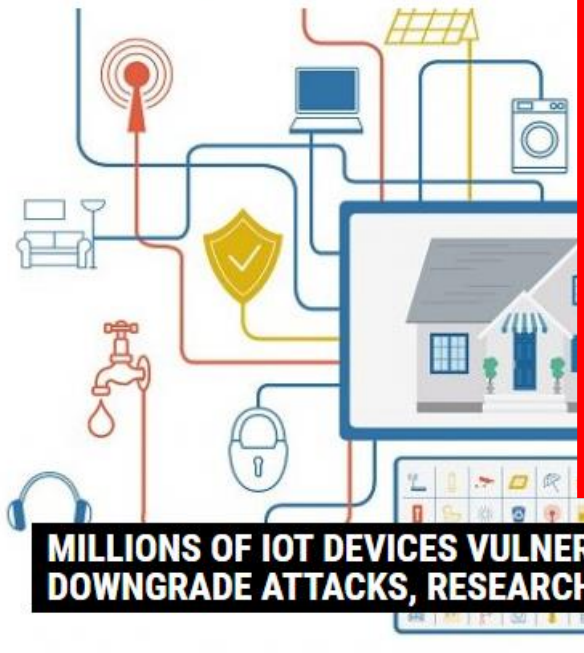
The Internet of Things bears little resemblance to traditional IT systems—and that makes it harder to protect

By **Stacey Higginbotham**

ZDNet

MUST READ: NEW AND UPCOMING PHONES: GALAXY NOTE

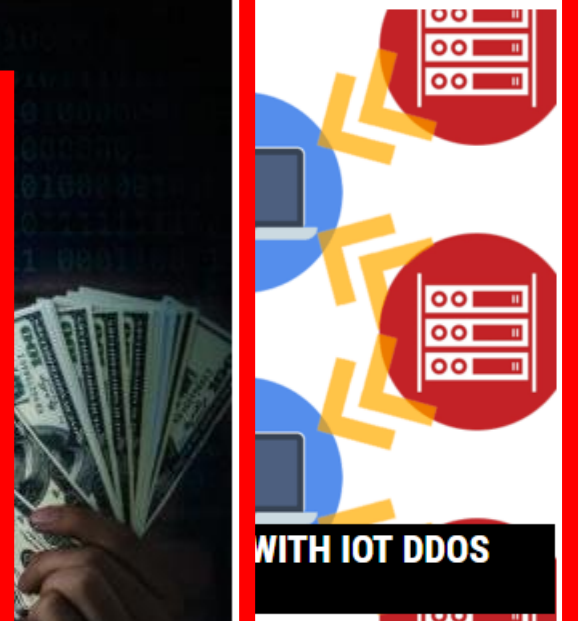
threat post



MILLIONS OF IOT DEVICES VULNERABLE TO Z-WAVE DOWNGRADE ATTACKS, RESEARCHERS CLAIM

by Tom Spring

May 25, 2018 , 3:27 pm



WITH IOT DDOS

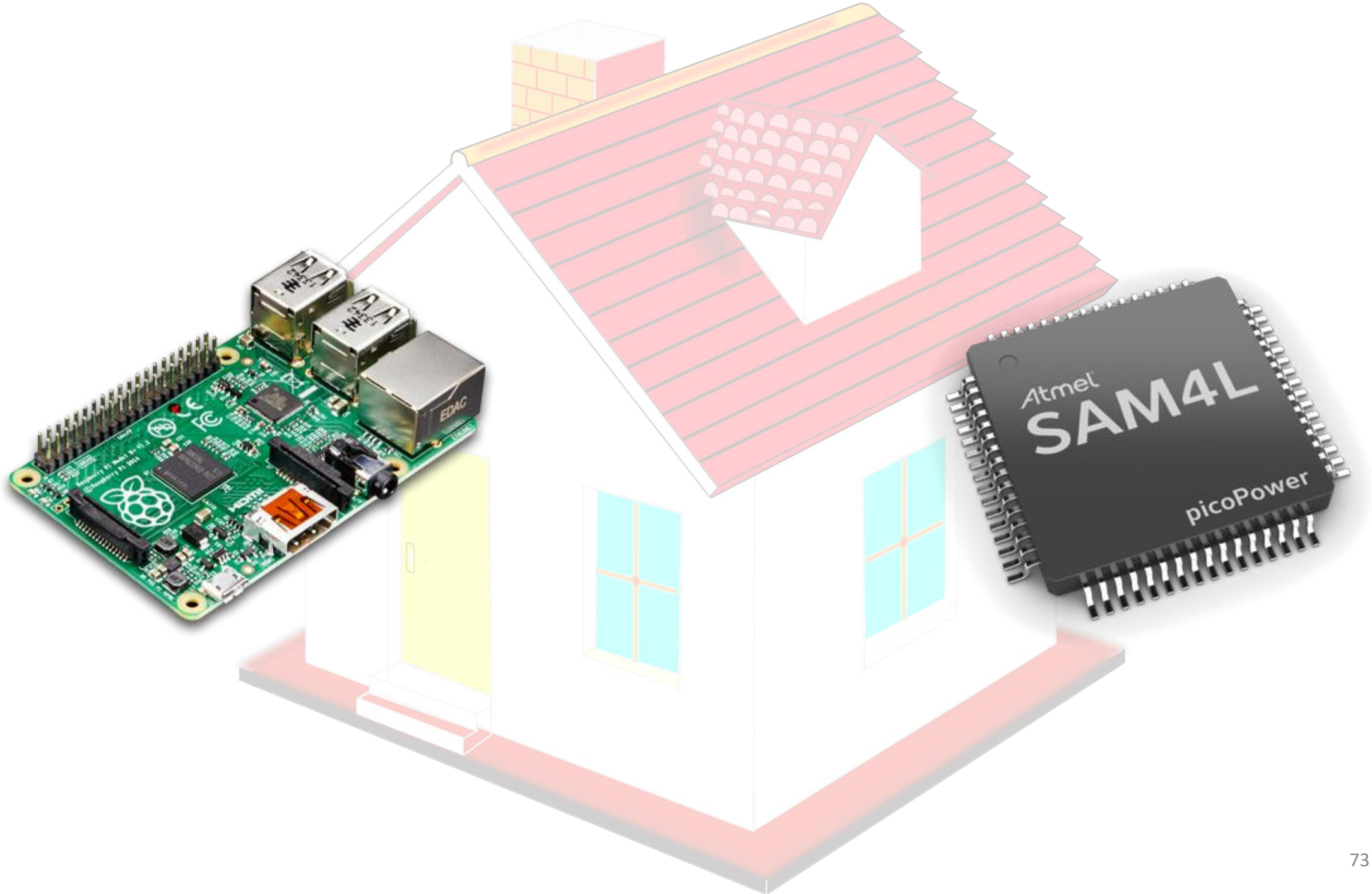
018 , 3:24 pm

ERVERS

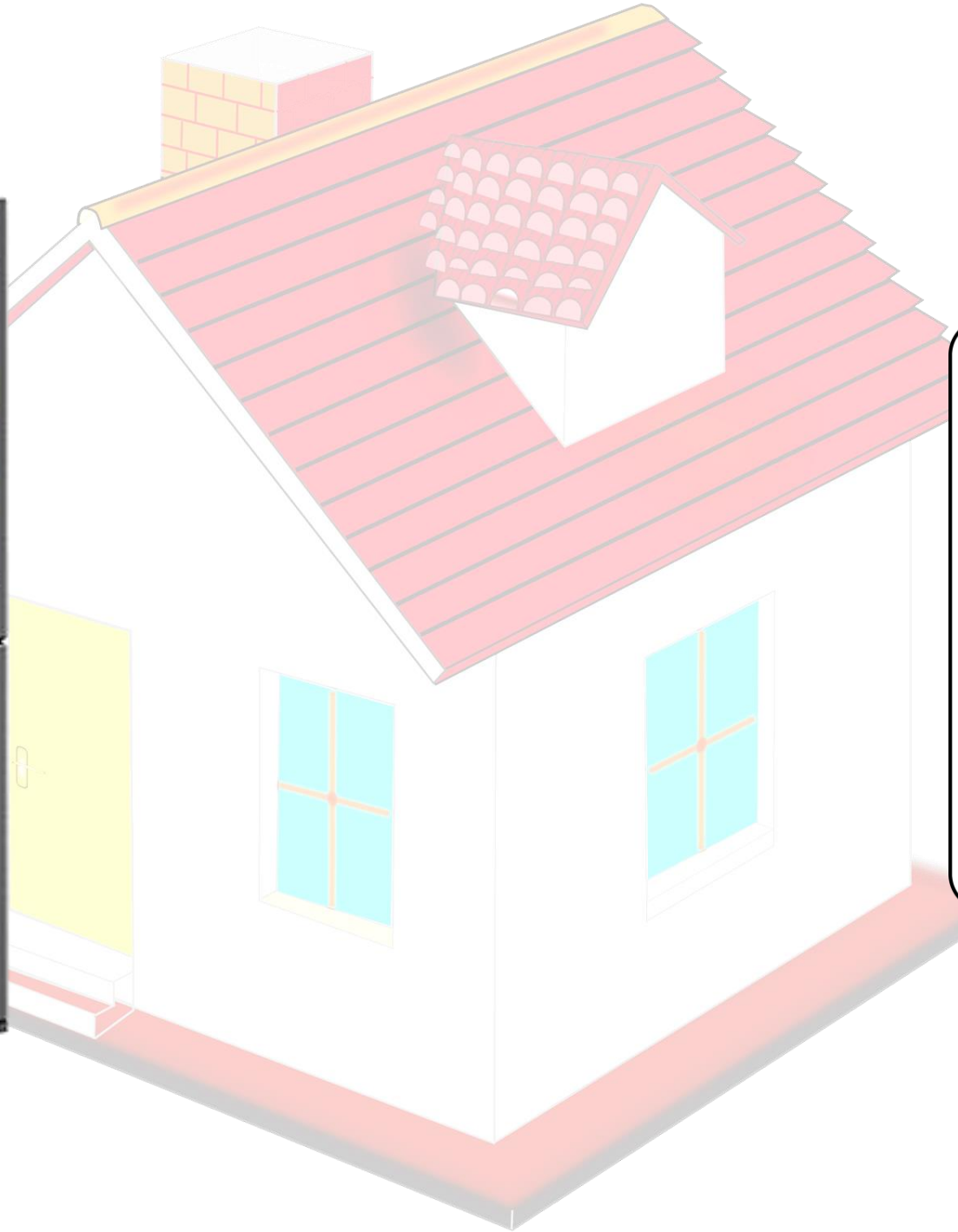
e 7, 2018 , 1:06 pm

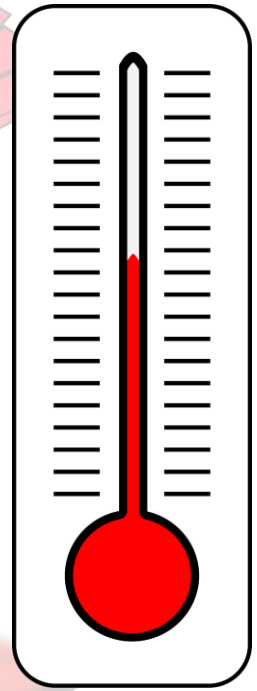
07 Study: Attack on KrebsOnSecurity Cost IoT Device Owners $323K

MAY 18


EDAC
Atmel
SAM4L
picoPower


EDAC
Power

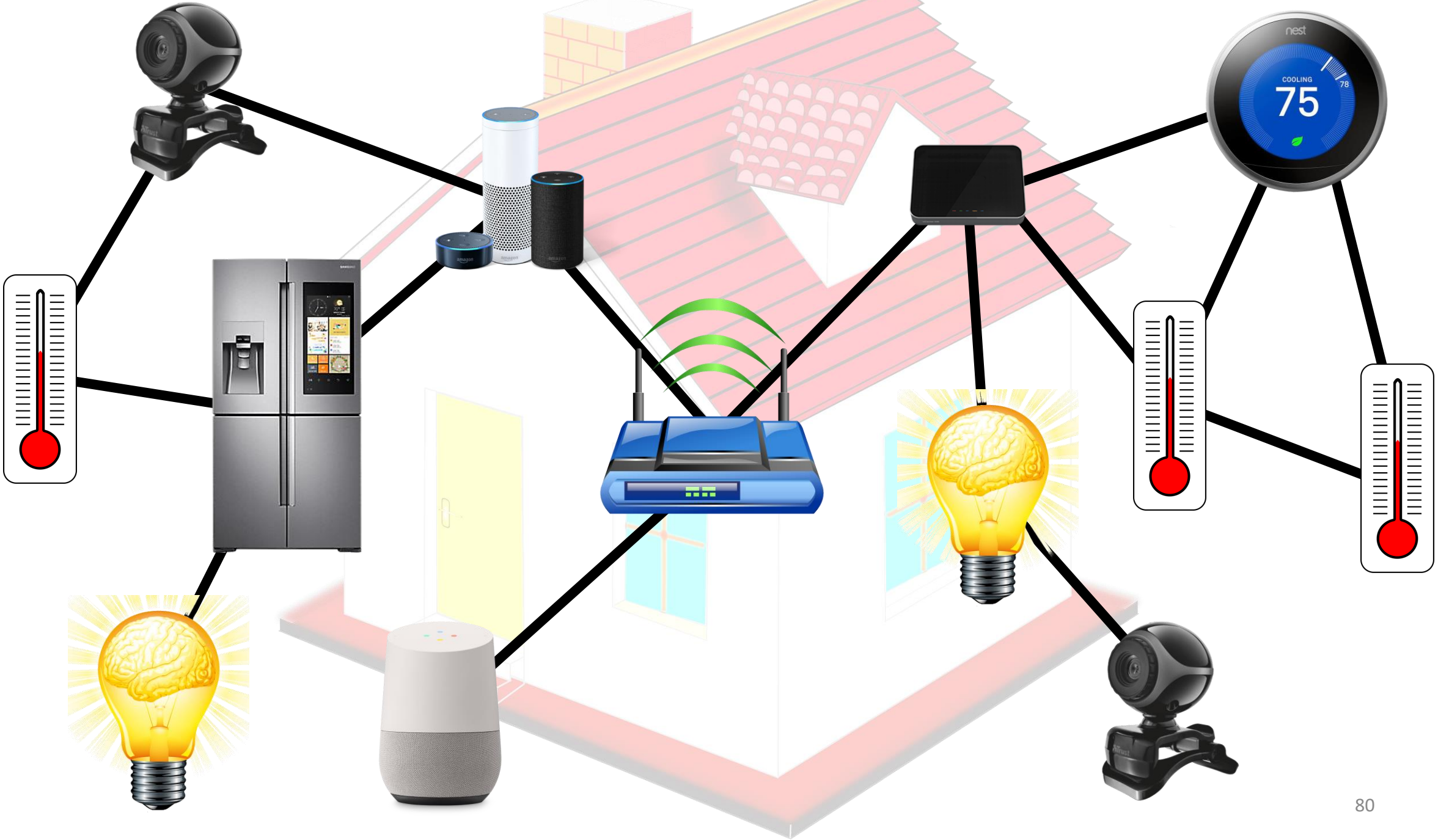


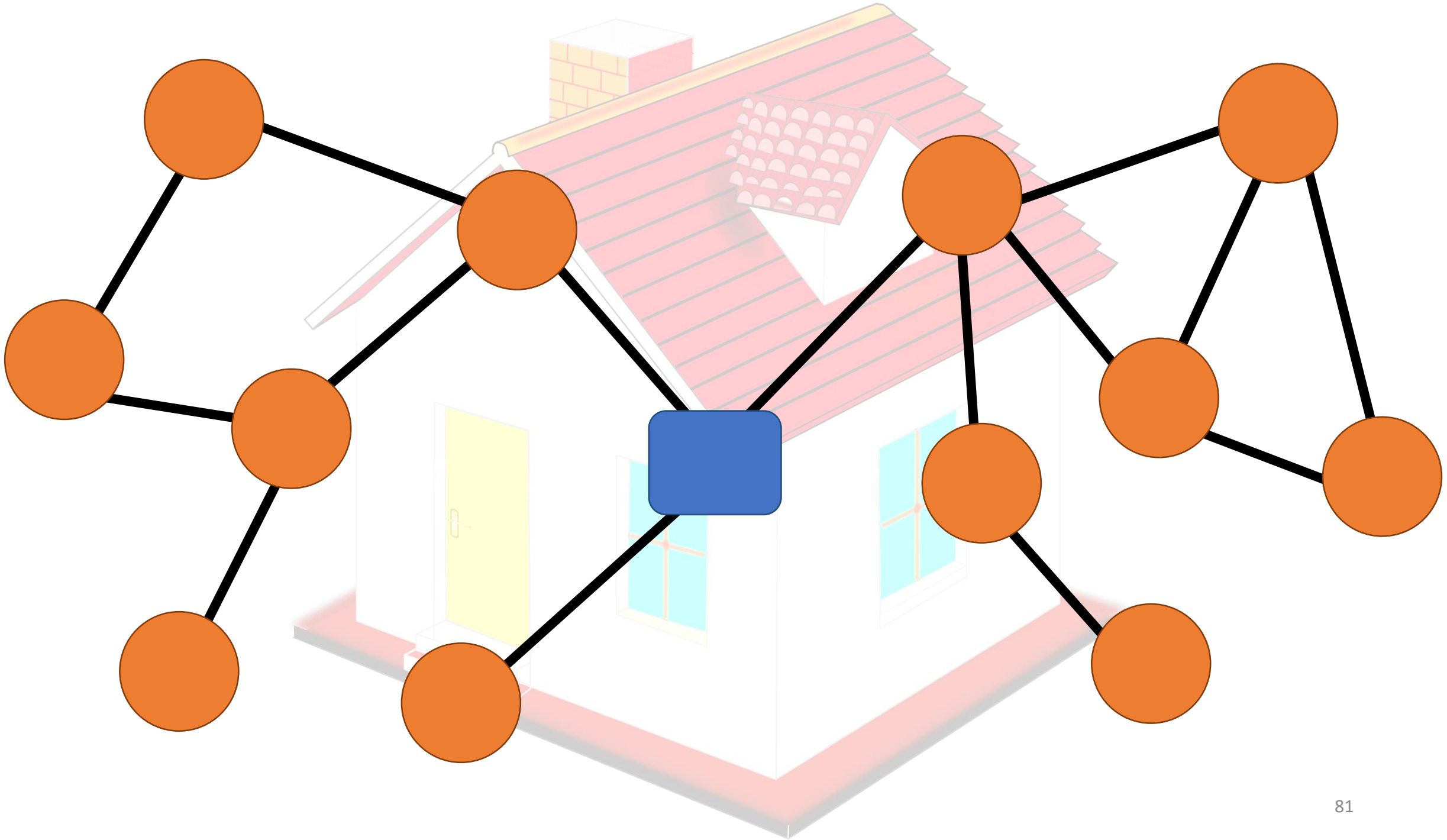


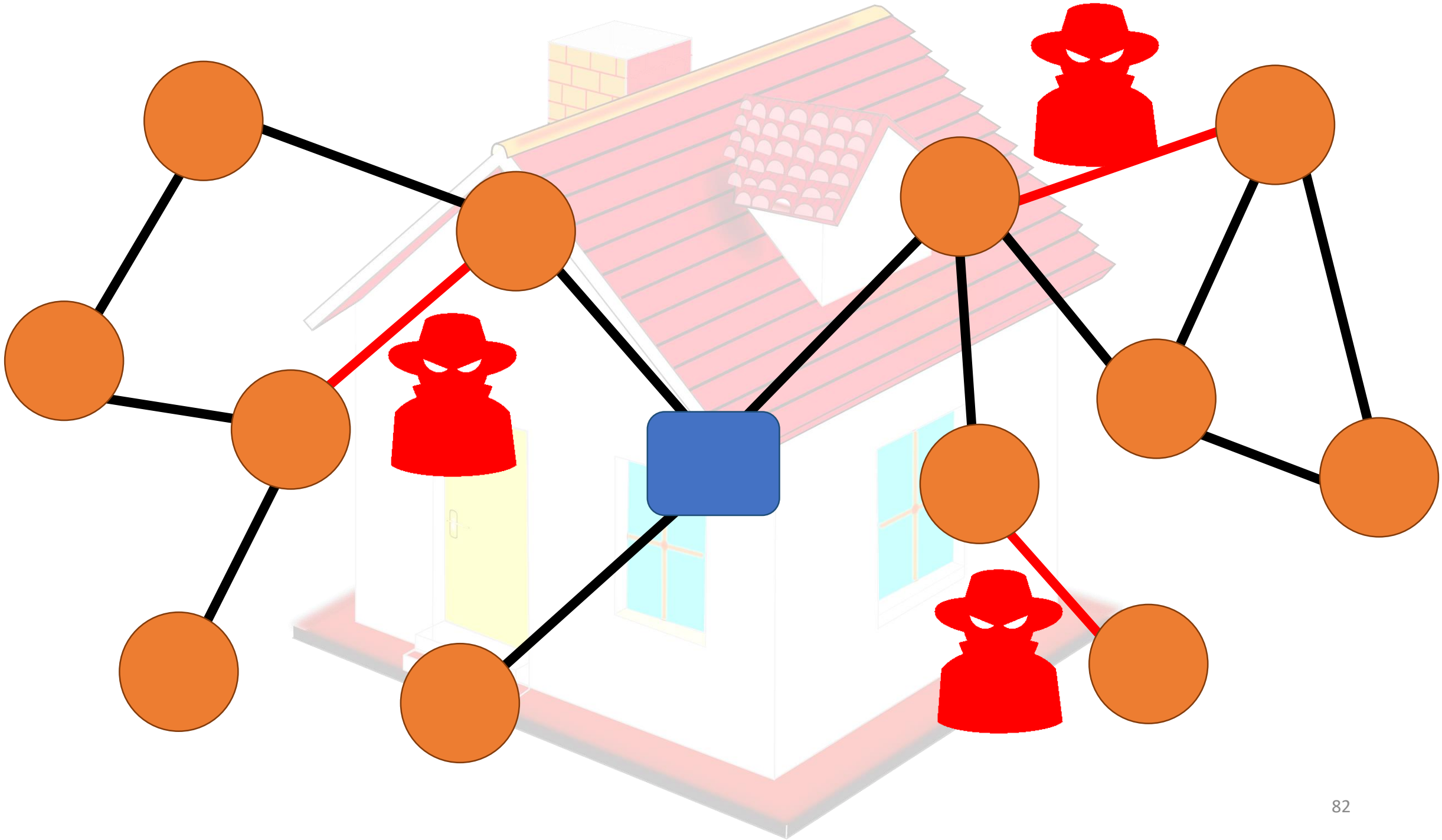

nest
COOLING
75
78


nest
COOLING
75
78


nest
COOLING
75
78


nest
COOLING
75
78





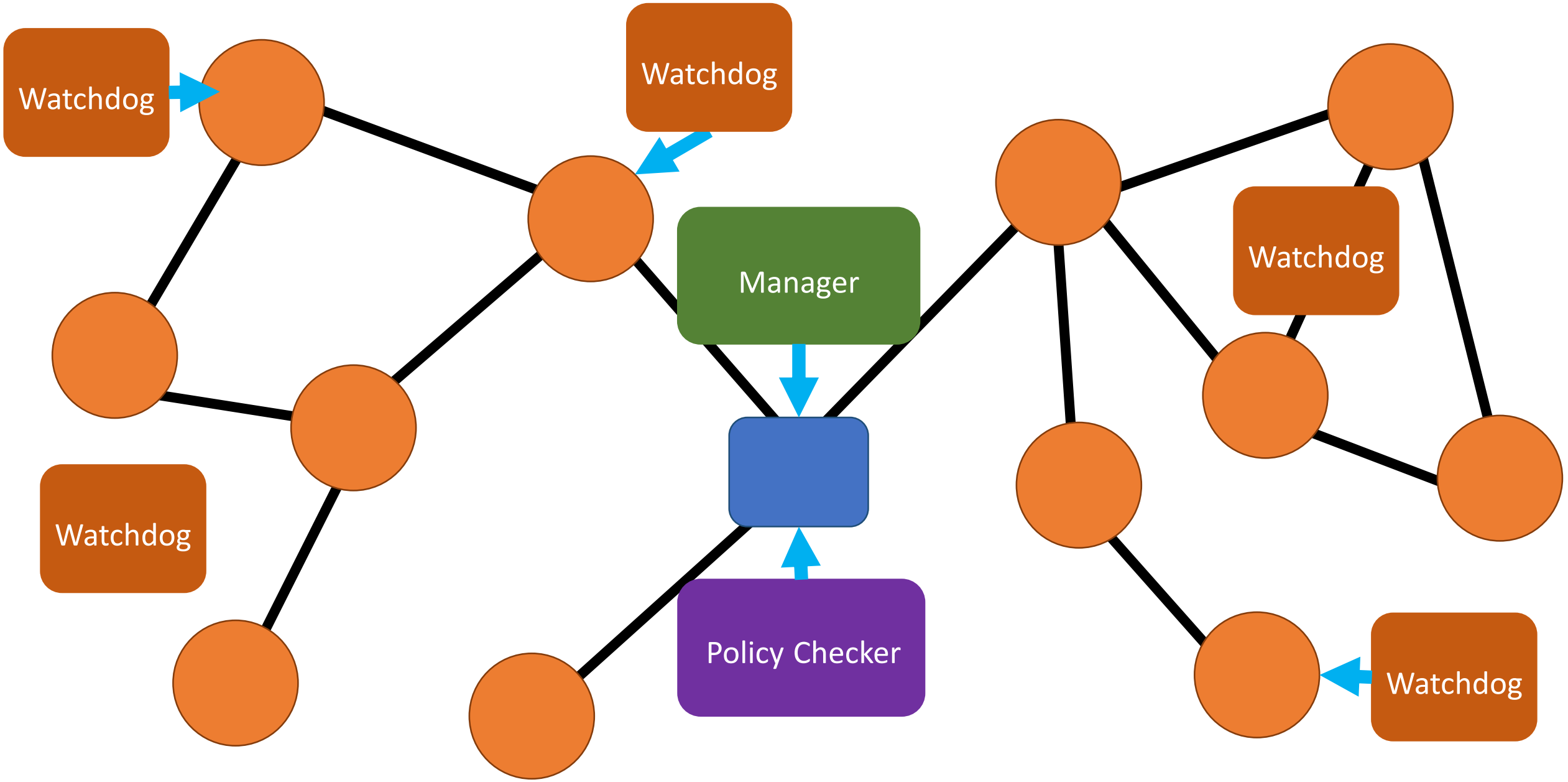
Related Work

Related Work

- H. Abie et al., “Risk-based adaptive security for smart IoT in eHealth,” in Proceedings of the 7th International Conference on Body Area Networks. 2012
- J. B. Bernabe et al., “Privacy-preserving security framework for a social-aware internet of things,” in International Conference on Ubiquitous Computing and Ambient Intelligence. 2014
- A. K. Simpson et al., ”Securing vulnerable home IoT devices with an in-hub security manager,” in IEEE International Conference on Pervasive Computing and Communications Workshops. 2017
- W. M. Kang et al., “An enhanced security framework for home appliances in smart home,” Humancentric Computing and Information Sciences. 2017

Related Work

- H. Abie et al., “Risk-based adaptive security for smart IoT in eHealth,” in Proceedings of the 7th International Conference on Body Area Networks. 2012
- J. B. Bernabe et al., “Privacy-preserving security framework for a social-aware internet of things,” in International Conference on Ubiquitous Computing and Ambient Intelligence. 2014
- A. K. Simpson et al., “Securing vulnerable home IoT devices with an in-hub security manager,” in IEEE International Conference on Pervasive Computing and Communications Workshops. 2017
- W. M. Kang et al., “An enhanced security framework for home appliances in smart home,” Humancentric Computing and Information Sciences. 2017


Watchdog
Watchdog
Manager
Watchdog
Watchdog
Policy Checker
Watchdog

Architecture

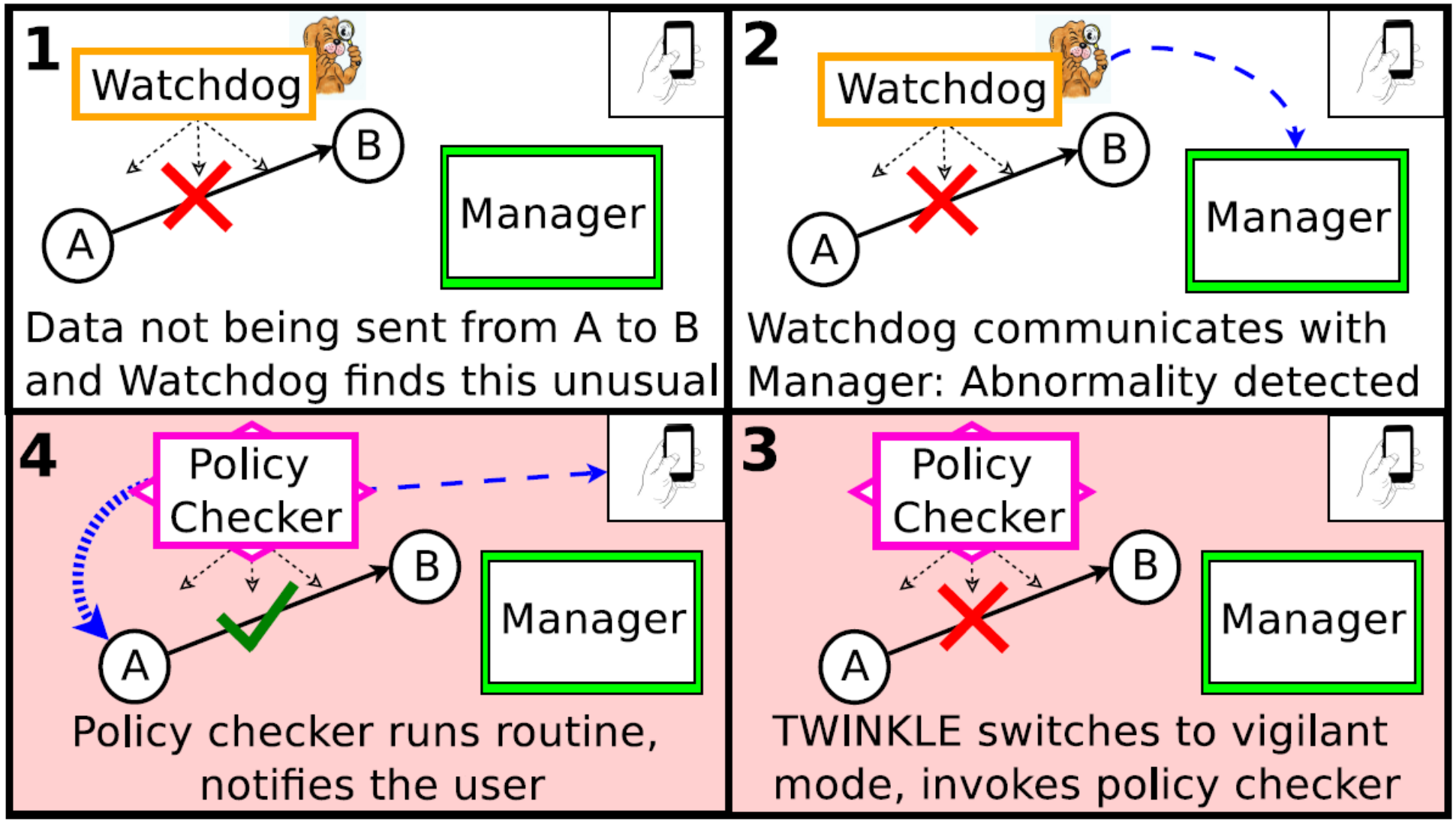
TWINKLE Architecture: Security Applications

- TWINKLE will instantiate manager, policy checker, and watchdog according to the security application
- Security application must:
  - Define the attack it targets and suspicious behavior that its watchdog should monitor
  - Develop routines to handle each suspicious behavior
  - Plug these routines into the policy checker
  - Populate the suspicious behavior handling table with every suspicious behavior that the security application is concerned about and the routine that handles the suspicious behavior

TWINKLE Architecture: Elf

- TWINKLE provides a dynamic mechanism for a security application to install its watchdog at any device needed
- Unlike the manager or policy checker that can run at a central node, depending on the security application in question, the watchdog may need to run on arbitrary devices in the smart home
- TWINKLE deploys a lightweight process called **elf** at each device that may be a candidate for running a watchdog
- TWINKLE can communicate with the elf on the device to ship, install, and eventually run watchdog code on the device

TWINKLE Handling a Jamming Attack



Case Study 1

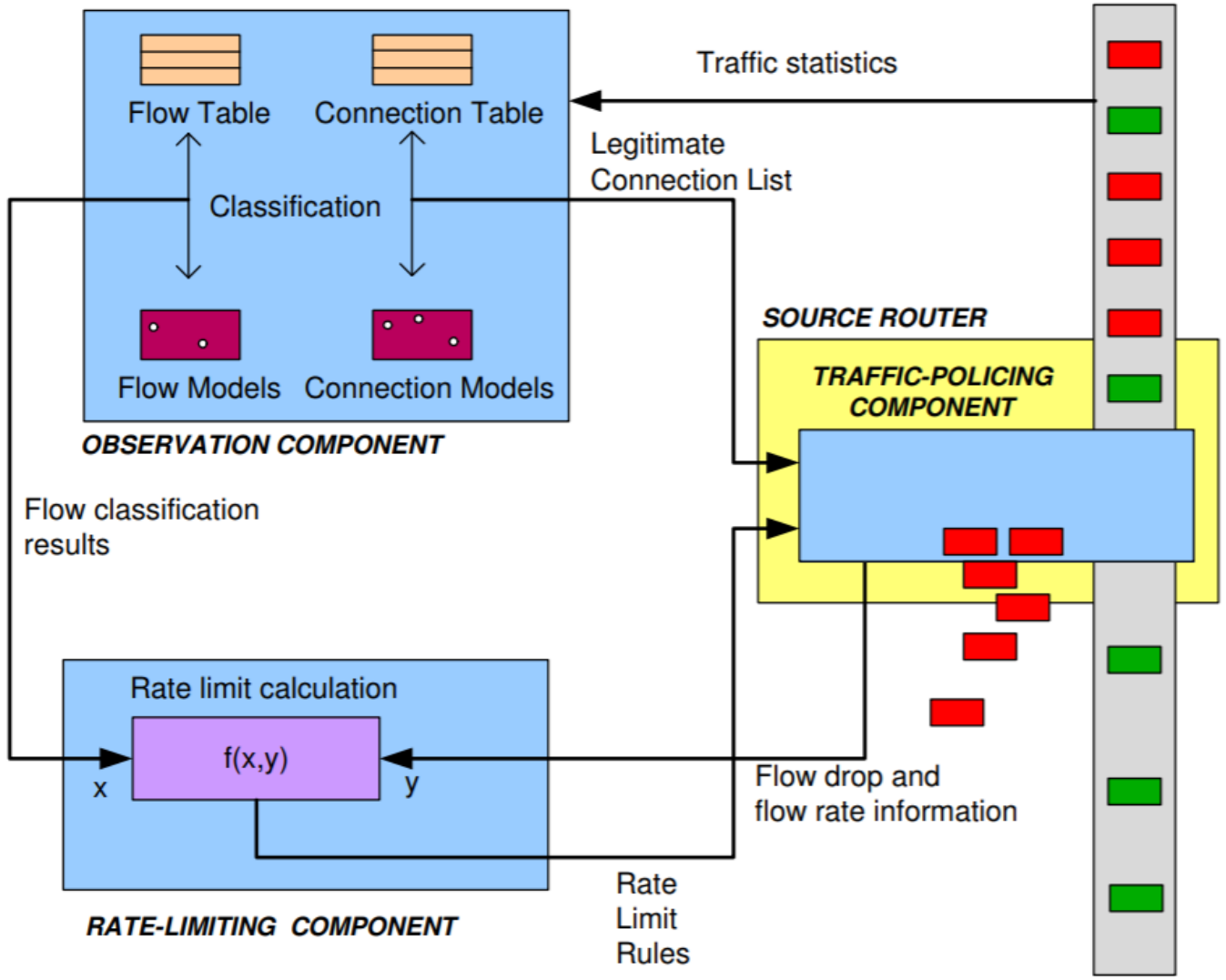
Prior Art: D-WARD Against DDoS Attacks

- Source-end solution – deployed at border router
- Classifies each aggregated flow (agflow) as good, suspicious, or attack
  - agflow – each connection from devices in the network to a specific destination outside the network
- Each connection in an attack agflow is classified as good, bad, and transient
- Classification is done based on ratio of sent packets to received packets
- Rate limits each bad and transient connection by dropping traffic to a fraction (f_{dec}) of the window size (W)
- If rate limit is followed for certain period of time, rate limit increases linearly and is eventually removed

Mirkovic, Jelena, and Peter Reiher. "D-WARD: A Source-End Defense Against Flooding Denial-of-Service Attacks." *IEEE Transactions on Dependable and Secure Computing* 2.3 (2005): 216-232.

Prior Art: D-WARD's Three Modules

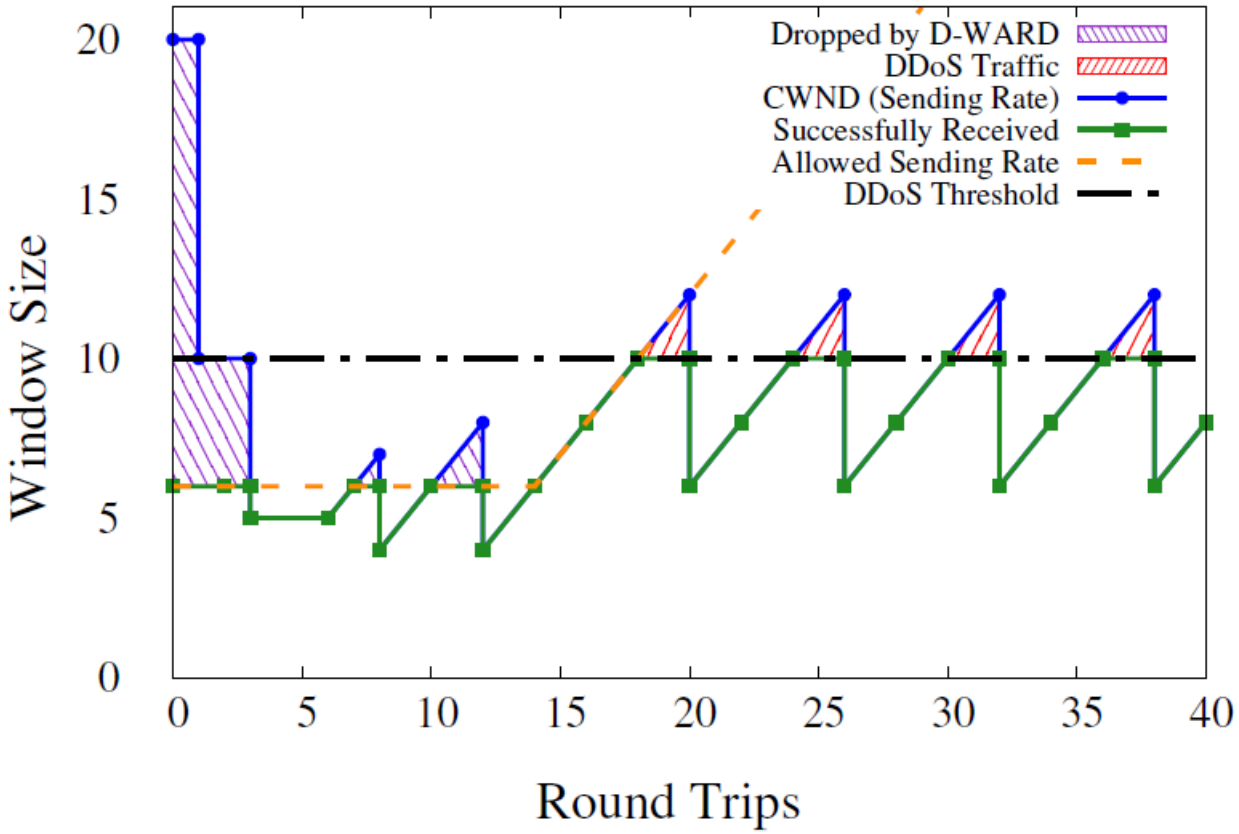
1. **Observation module**: classifies each agflow and each connection in a bad or transient agflow
2. **Rate-limiting module**: calculates and updates rate-limit
3. **Traffic-policing module**: drops all traffic surpassing the rate-limit



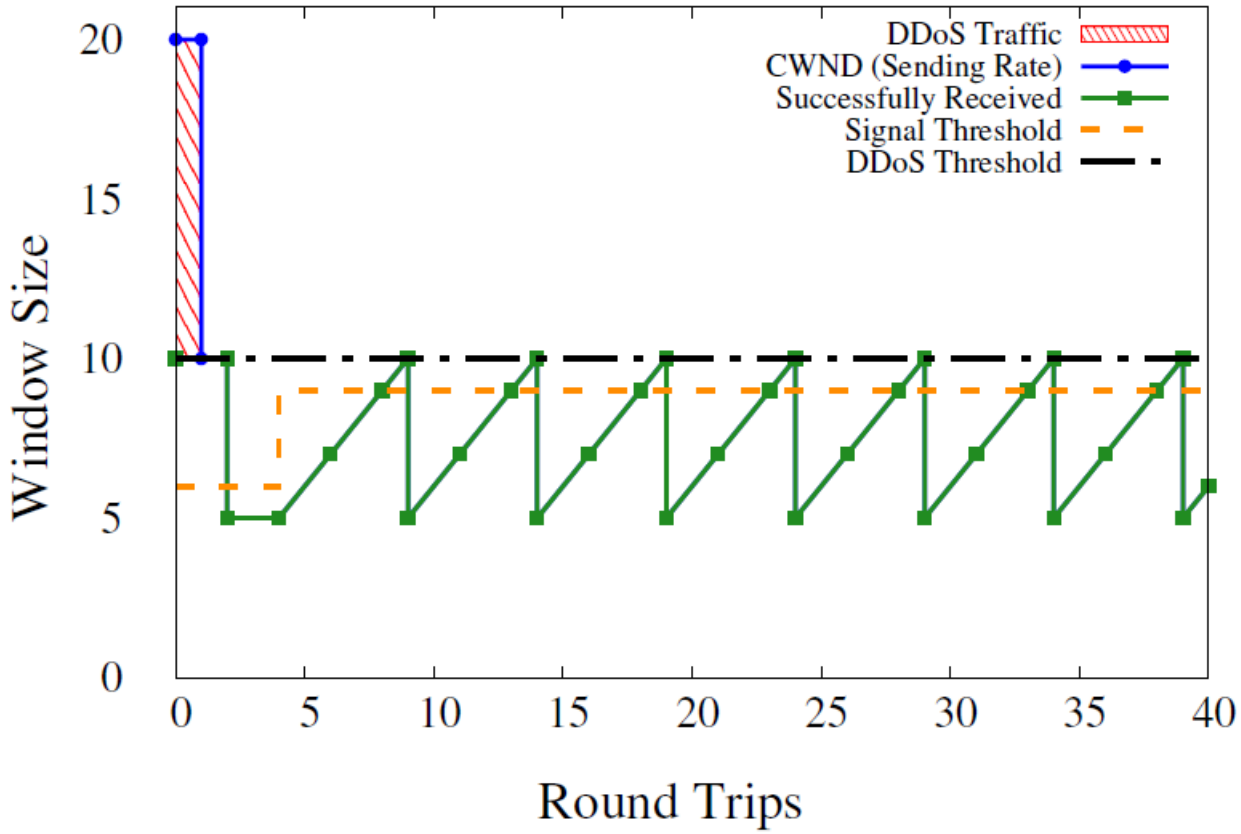
D-WARD+: A Two-Mode Approach Against DDoS Attacks

- Manager: keeps track of rate-limit of every connection in every attack aggregated flows or **agflows** (equivalent to rate-limiting module)
- Policy checker: consists of an agflow monitoring routine (equivalent to observation module for connections and traffic-policing module)
- Watchdog: monitors the suspicious behavior of each agflow and has the agflow monitoring routine invoked if it detects an attack agflow (equivalent to observation module for agflows)
- All components are installed on the border router

Effect on “Smart” Attacker

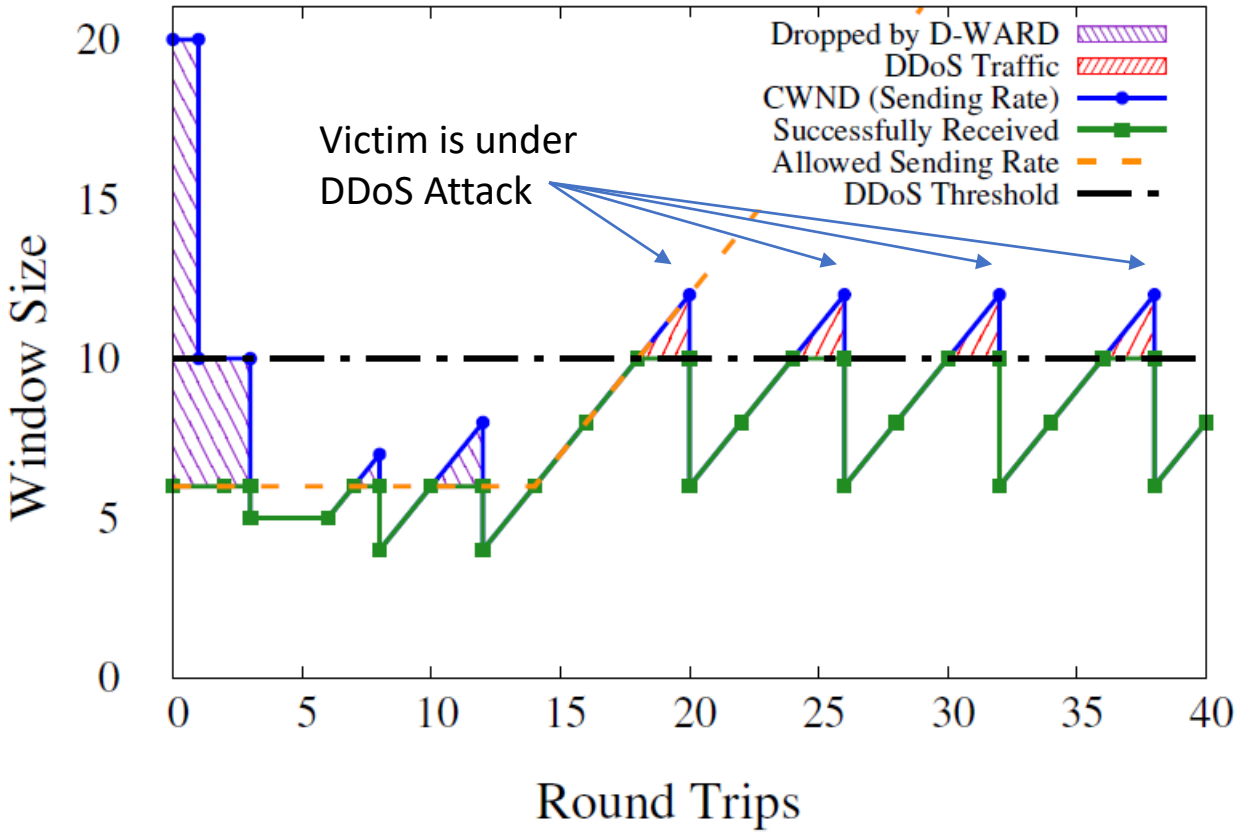


(a) D-WARD

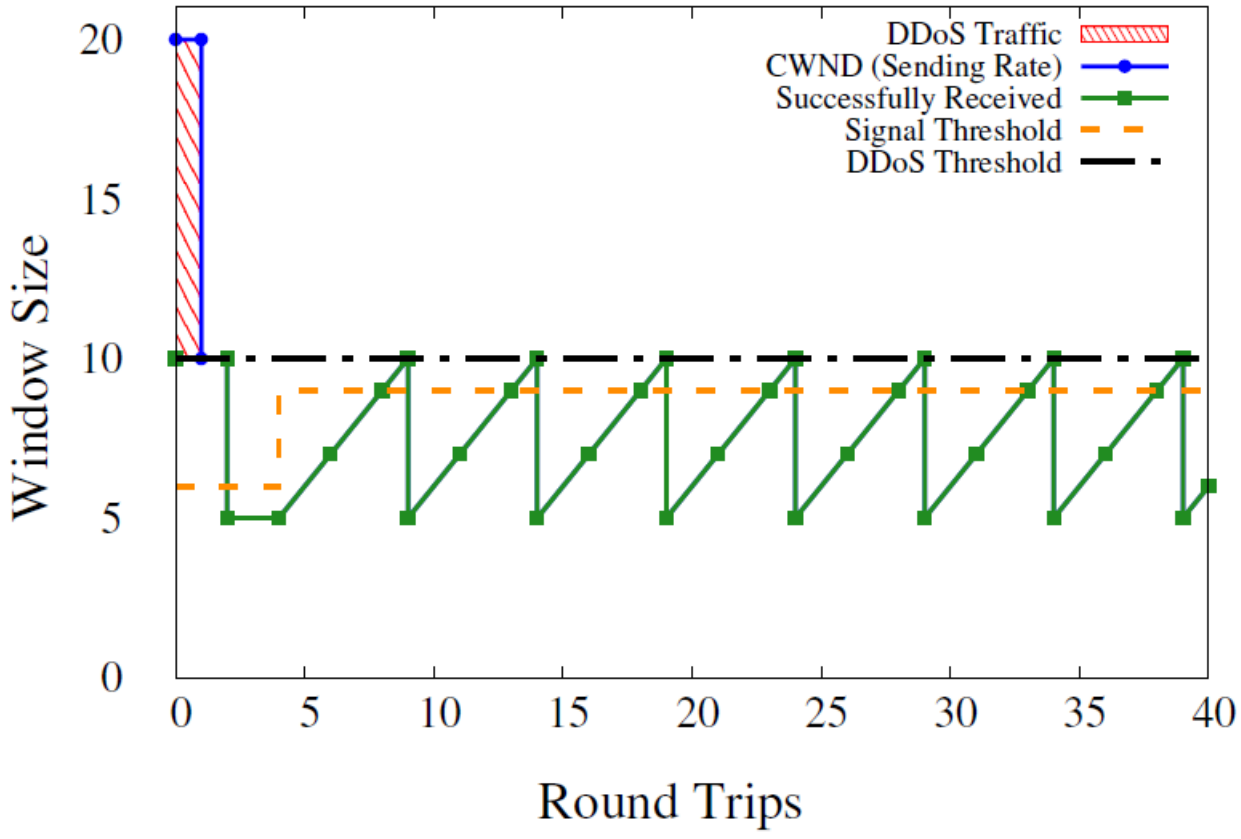


(b) D-WARD+

Effect on “Smart” Attacker



(a) D-WARD



(b) D-WARD+

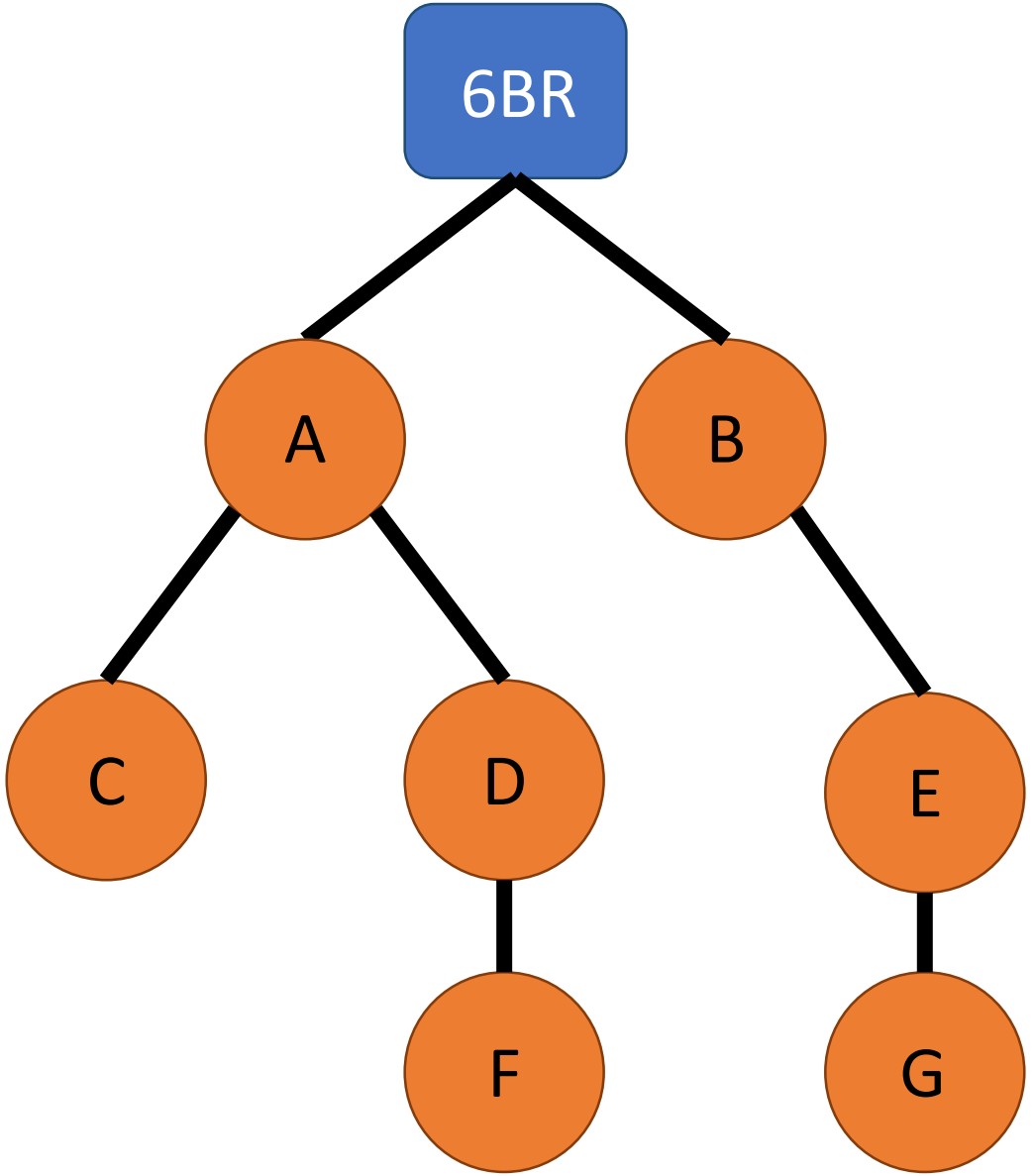
In Conclusion

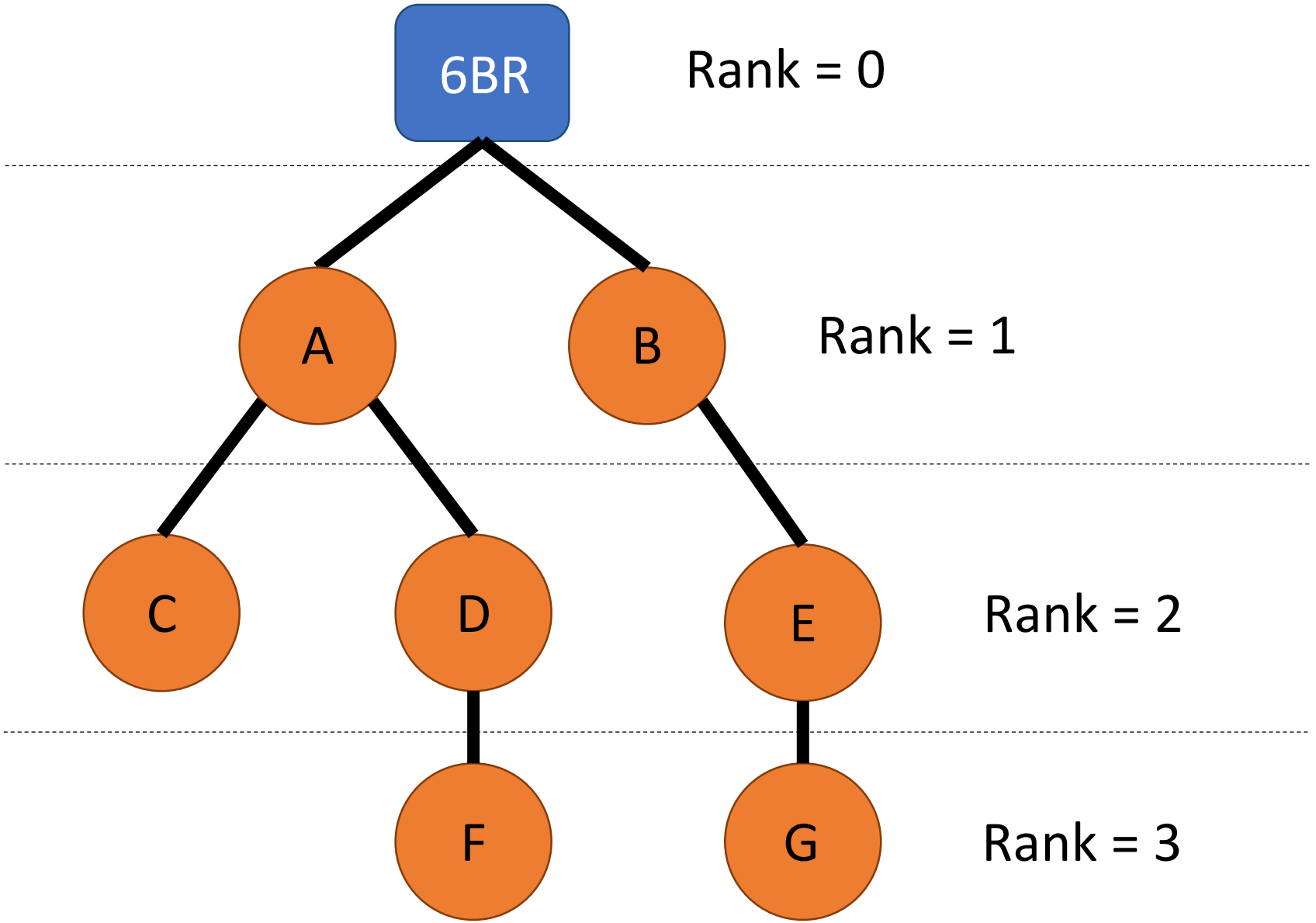
Based on the two-mode design, D-WARD+ is more suitable to a smart home environment than D-WARD. By not literally dropping packets as in D-WARD, D-WARD+ instead informs devices to transmit more slowly. Doing so avoids retransmissions of packets from benign devices, thus lowering network overhead and power consumption.

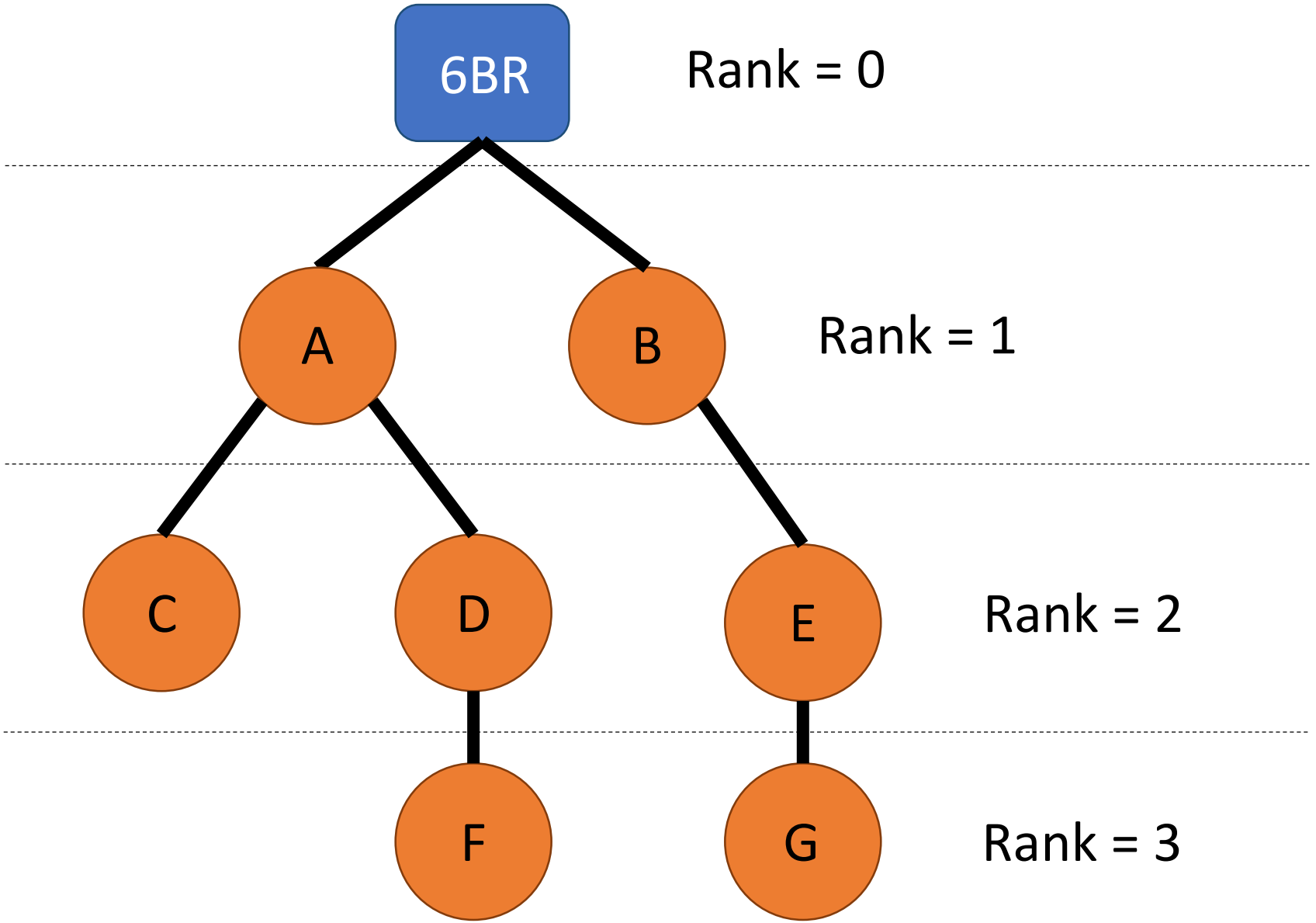
Case Study 2

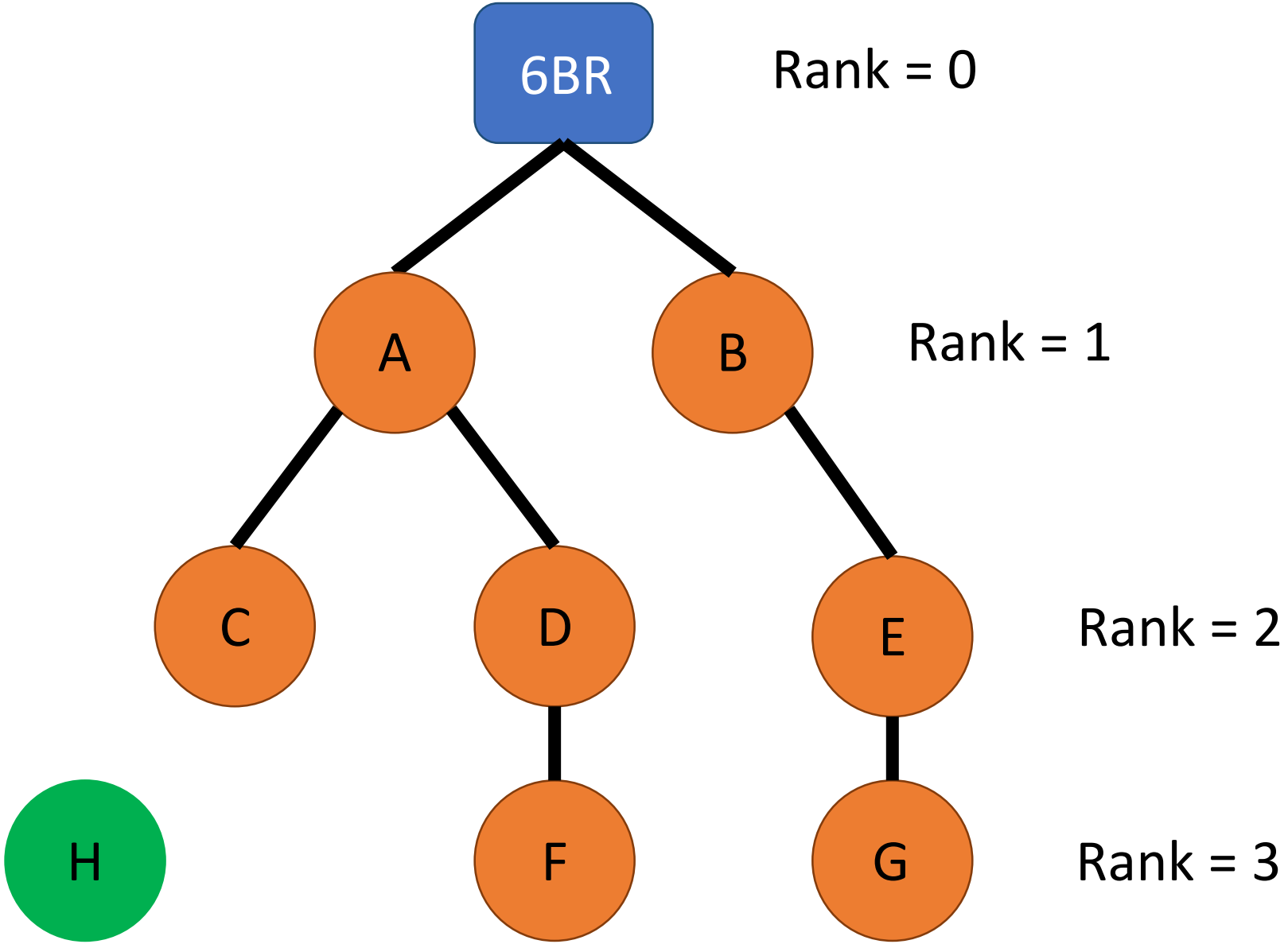
6LoWPAN Networks

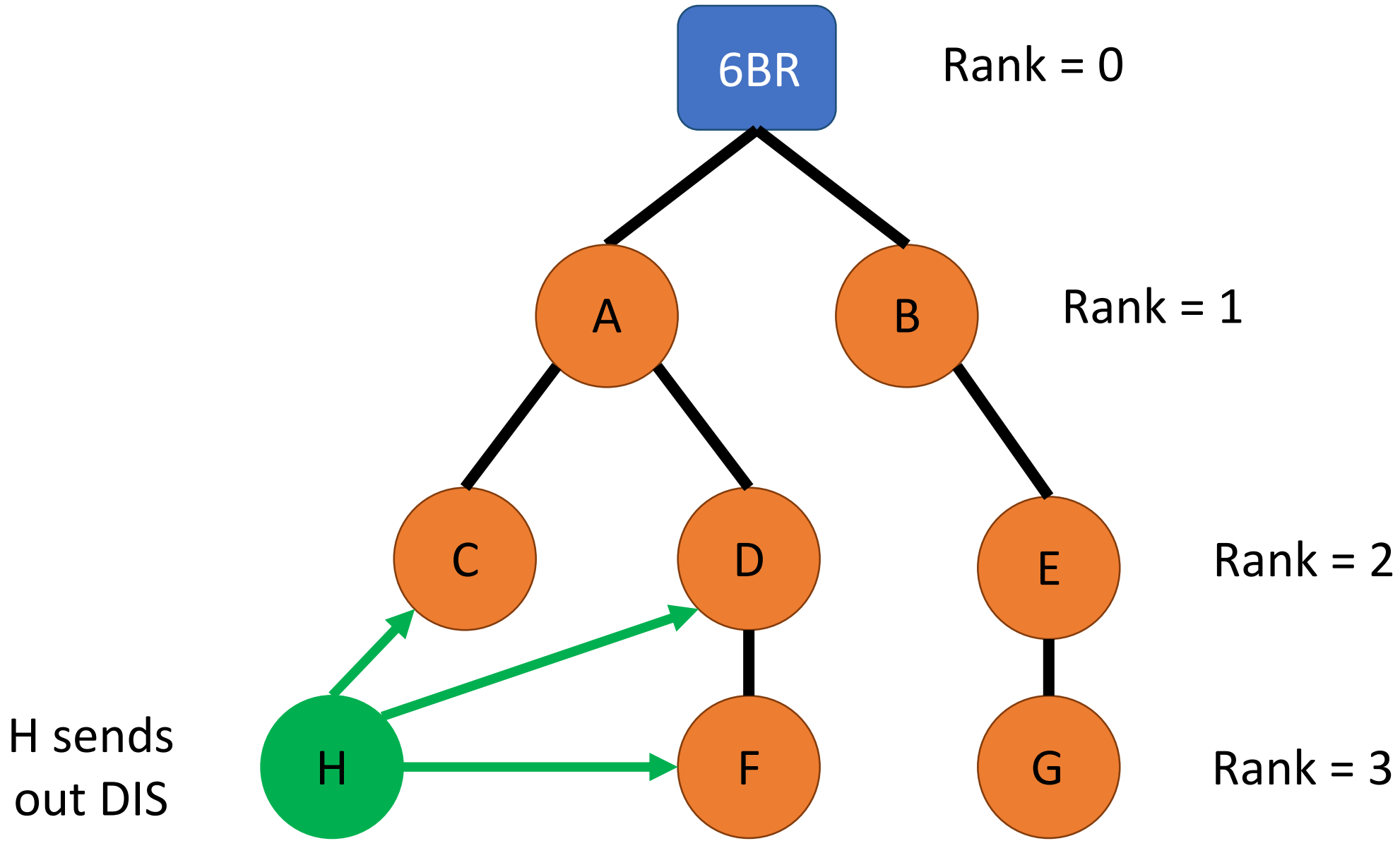
- 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks): a wireless technology that combines IPv6 and Low-power Wireless Personal Area Networks (LoWPAN) to enable low-powered devices to communicate using an Internet protocol
- 6LoWPAN uses RPL (Routing Protocol over Low Powered and Lossy Networks) as its routing protocol


6BR
A
B
C
D
E
F
G

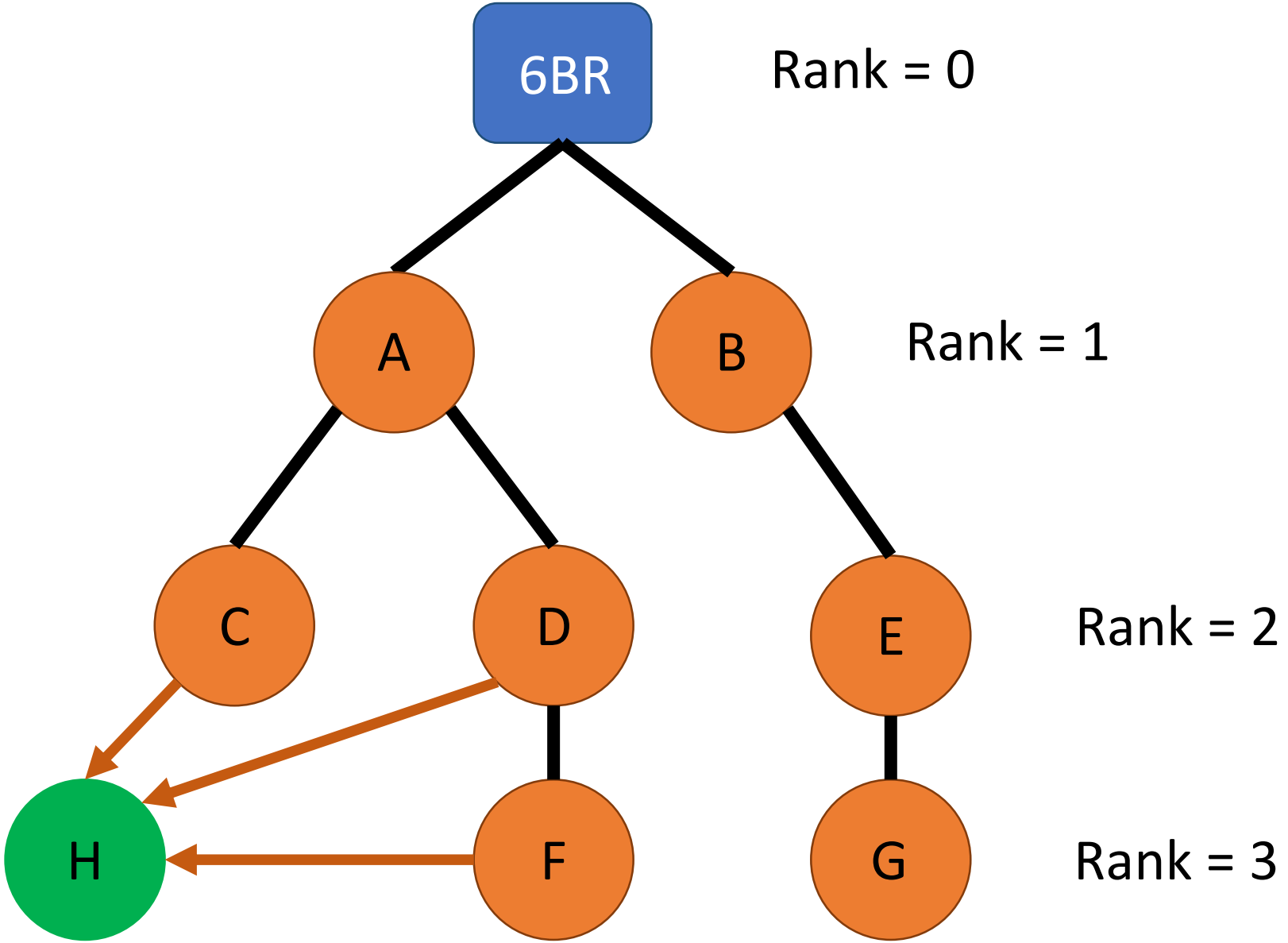

6BR
Rank = 0
A
B
Rank = 1
C
D
E
Rank = 2
F
G
Rank = 3

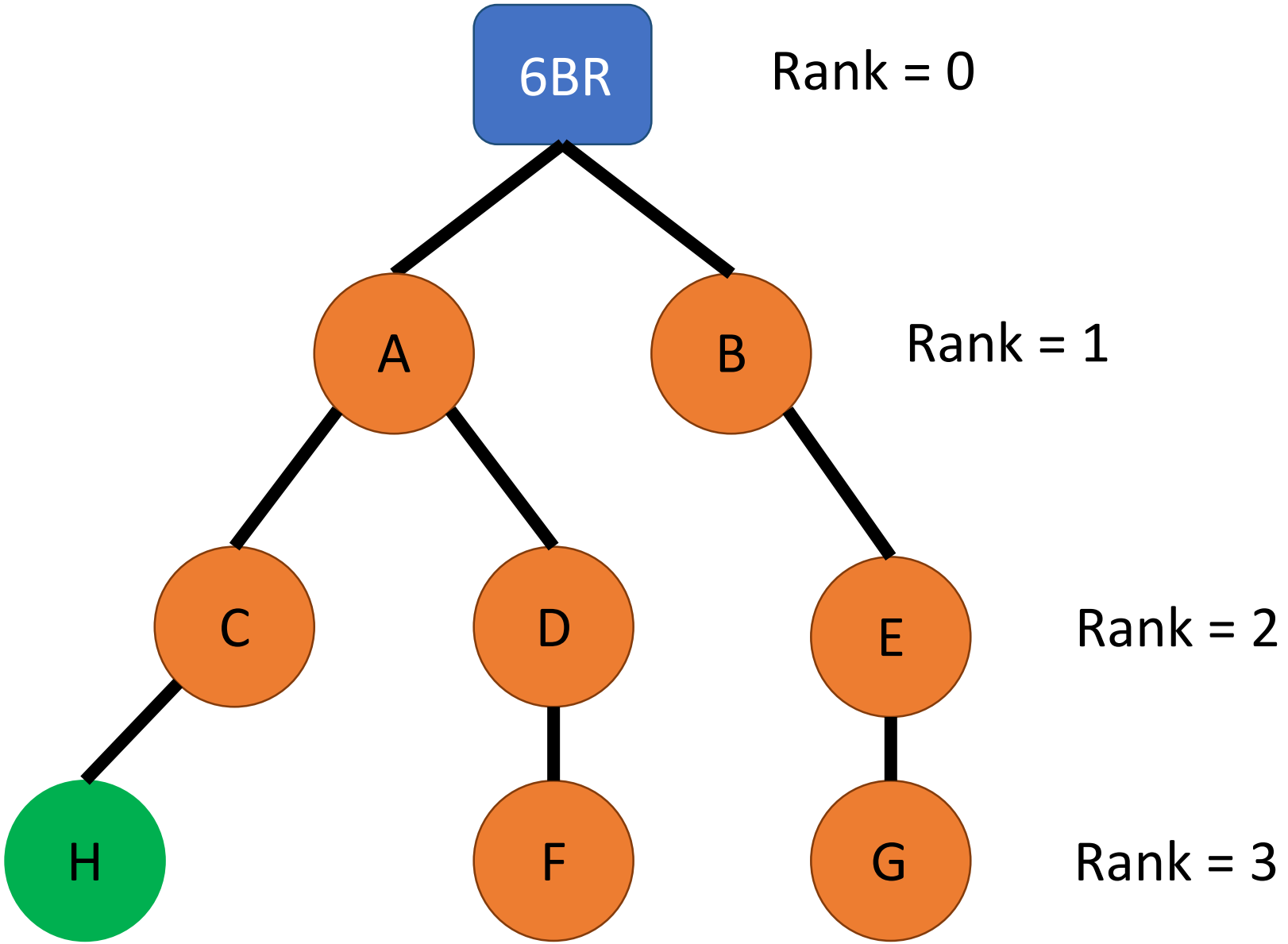

6BR
Rank = 0
A
B
Rank = 1
C
D
E
Rank = 2
F
G
Rank = 3


6BR
Rank = 0
A
B
Rank = 1
C
D
E
Rank = 2
H
F
G
Rank = 3

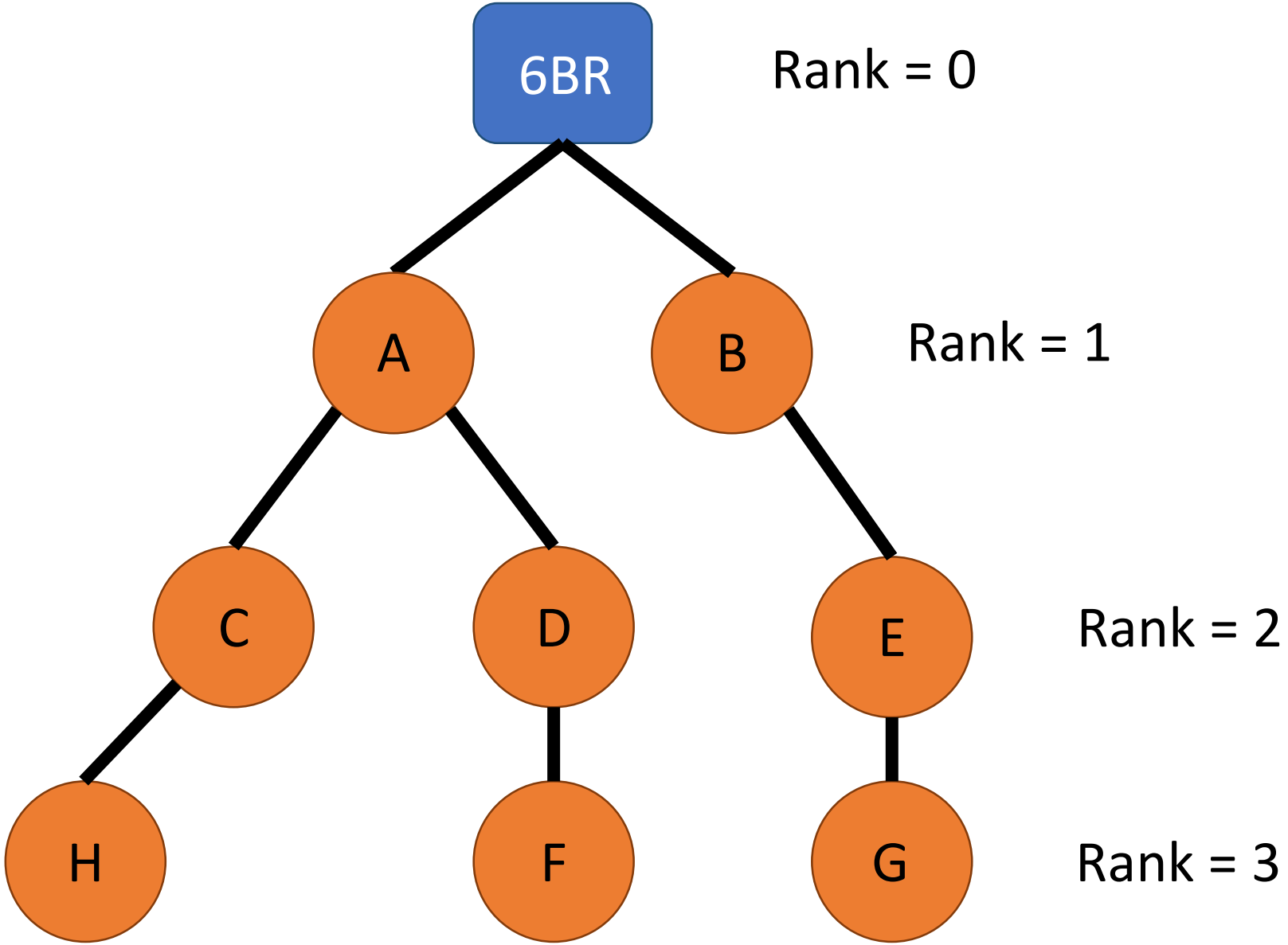

6BR
Rank = 0
A
B
Rank = 1
C
D
E
Rank = 2
H
F
G
Rank = 3
H sends
out DIS

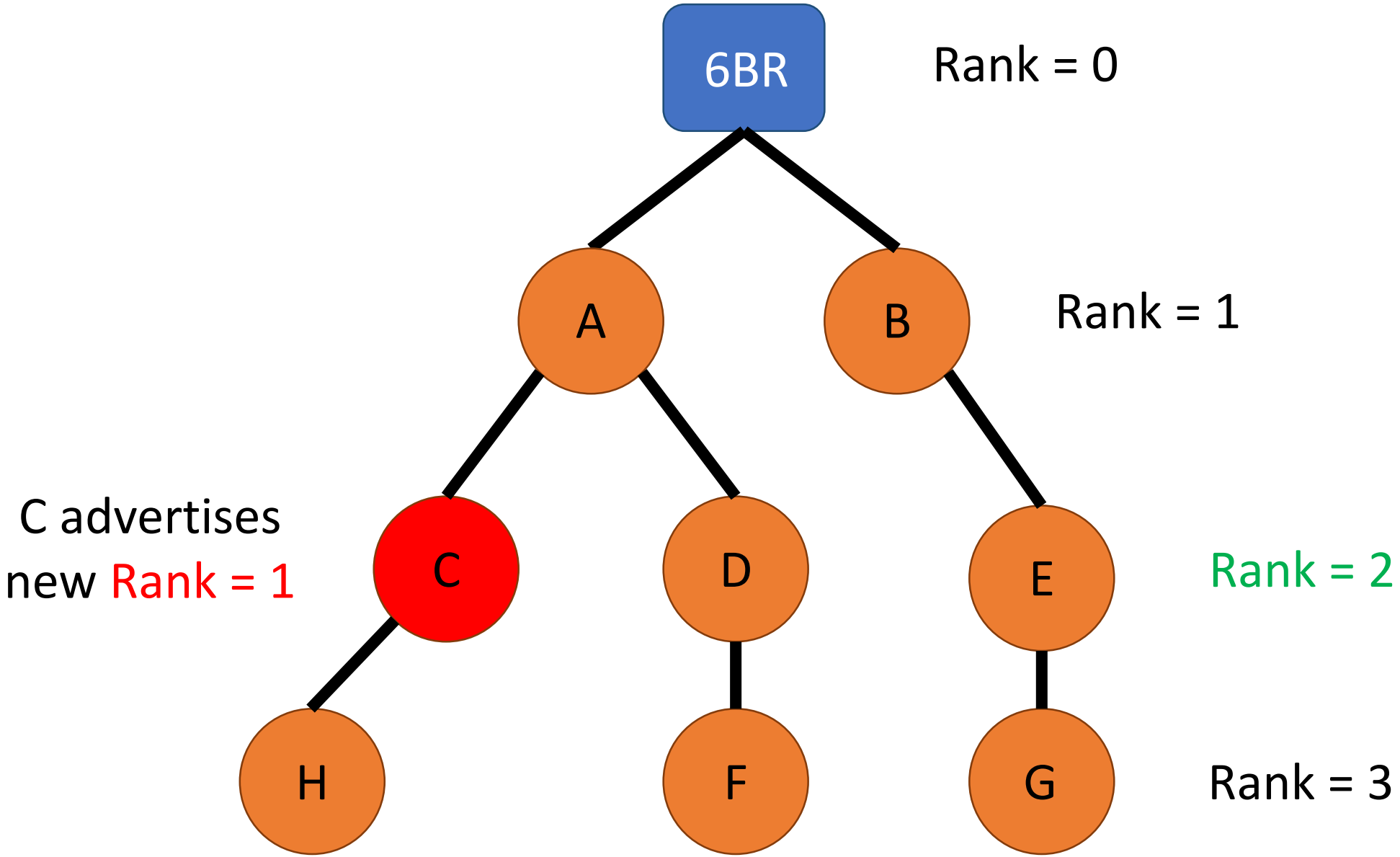
C, D, F send H DIO (includes their rank)

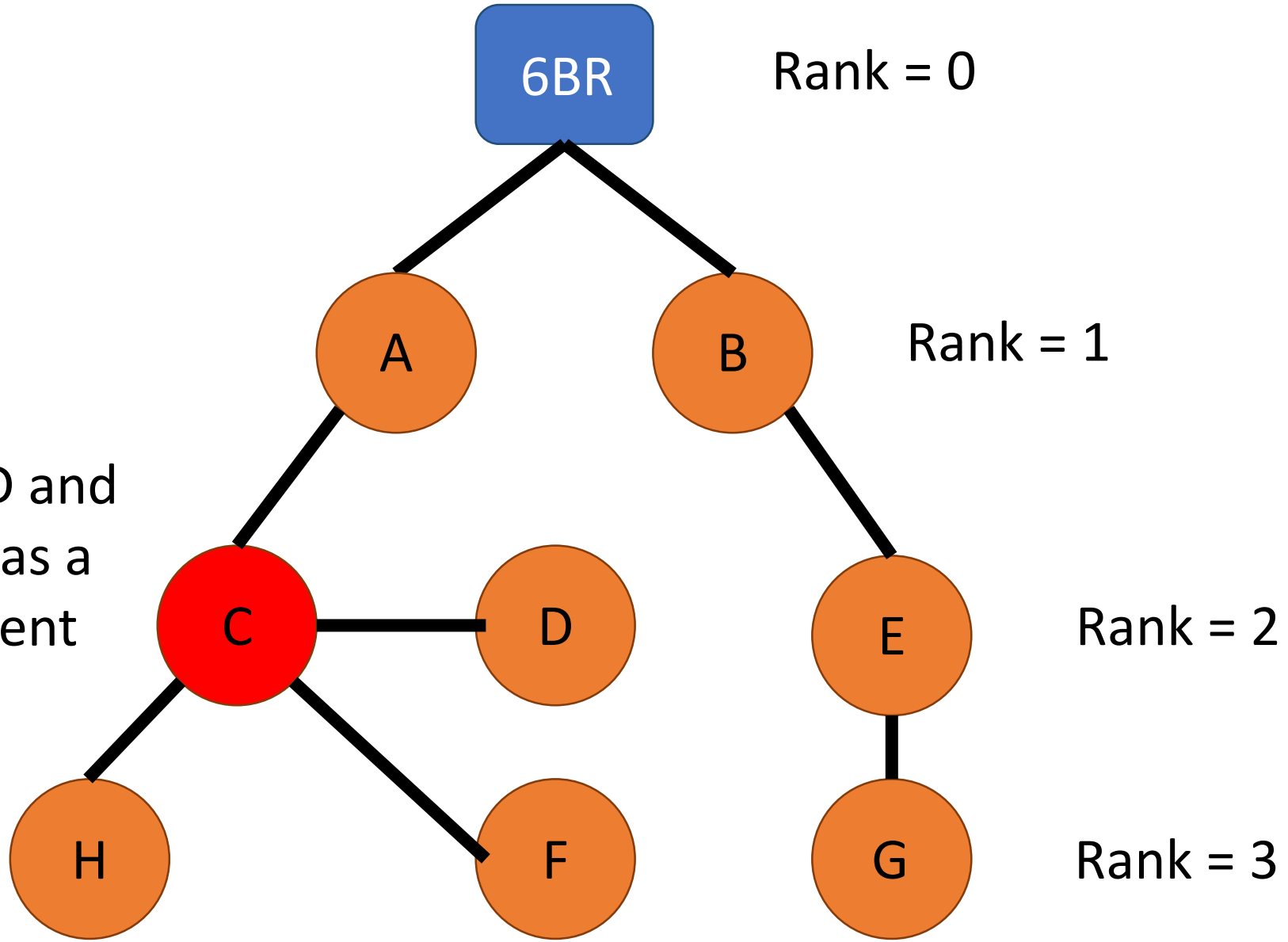




H creates parent set (C, D) and chooses preferred parent (C)


6BR
Rank = 0
A
B
Rank = 1
C
D
E
Rank = 2
H
F
G
Rank = 3


6BR
Rank = 0
A
B
Rank = 1
C advertises
new Rank = 1
C
D
E
Rank = 2
H
F
G
Rank = 3


6BR
Rank = 0
A
B
Rank = 1
Which causes D and F to choose it as a preferred parent
C
D
E
Rank = 2
H
F
G
Rank = 3

Prior Art: SVELTE Against the Sinkhole Attack in 6LoWPAN

- Two main modules running on the border router (6BR) of a 6LoWPAN network:
  1. 6LoWPAN Mapper (6Mapper): gathers information about the network and determines the routing map (DODAG rooted at 6BR)
  2. Intrusion detection module: checks the rank inconsistency in data obtained by 6Mapper to detect sinkhole attacks
- 6Mapper sends probing messages to all nodes in network at regular intervals (e.g., 2 minutes)
- Each node then sends a response message to 6Mapper, which includes its node ID, node rank, parent ID, and all of its neighbors' IDs and ranks.

Raza, Shahid, Linus Wallgren, and Thiemo Voigt. "SVELTE: Real-time intrusion detection in the Internet of Things." *Ad Hoc Networks* 11.8 (2013): 2661-2674.

SVELTE+: A Two-Mode Approach Against 6LowPAN Sinkhole Attacks

- Manager (6BR): 6Mapper module from SVELTE
- Policy checker (6BR): two suspicious behavior handling routines
  1. Sinkhole detection routine (i.e., the intrusion detection module from SVELTE) that inspects the ranks of nodes in the routing map to determine if a sinkhole attack is occurring
  2. Sinkhole mitigation routine that SVELTE+ newly introduced to mitigate a detected sinkhole attack
- Watchdog (device): monitors the ranks of its neighbors and alerts the manager of a suspicious behavior when it receives a new rank advertisement

In Conclusion

SVELTE+ can reduce the latency in detecting sinkhole attacks to a negligible amount because the watchdog immediately invokes the suspicious behavior handler whenever a new rank is advertised, without having to wait for the next probing interval, as in SVELTE. SVELTE+ also decreases the network overhead and device power consumption as compared to SVELTE as a result of on-demand probing versus periodic probing.