communicates with IoT devices through the network con-
states. For example, the loss of a presence-off event could
nection layer that uses various communication techniques
leave the door unlocked after the resident leaves home.
such as WiFi, Zigbee, and ZWave. An IoT devices can be
Command Failures. They correspond to commands issued
partitioned into the cyber part and the physical part. The
by the IoT platforms that fail to be executed by the target
cyber part manages interfaces for humans and bridges the
devices. Command failures may be caused by malfunctions
communication between the cloud and the physical part, and
of a cyber part or physical part. (1) Cyber-part malfunc-
the latter fulfills its functions in the physical world. Taking
tions that cause commands to fail to execute, such as system
the Philips' Hue smart light bulb as an example, the physical
crashes and unstable network connections, are considered
part is the LED light bulb and the cyber part is the embedded
in our work. For example, the TP-Link smart plug often goes
micro-controller with a built-in wireless component.
irresponsive [11]. (2) A physical-part malfunction is equiv-
Next, we describe some terms used in SmartThings. A
alent to a malfunction in a traditional (i.e., non-smart) device.
device has one or multiple capabilities, each categorized as
For example, a broken electrical relay inside a smart plug
an actuator or sensor. Each capability defines one or more
can prevent the plug from cutting off the power supply [18],

attributes. For example, a smart plug device has an attribute
although from the perspective of the IoT platform, the plug
"switch" and, optionally, an attribute "power." Each attribute's
has been turned off.
state (i.e., value) is stored on the cloud and updated due to
events sent from the IoT device. For example, the Smart-
3.2 Attacks on IoT Devices
Things multipurpose sensor has a capability contact sensor,
We survey the recent work on attacks against IoT devices,
whose attribute "contact" changes from "open" to "closed"
and find HAWatcher has the potential to detect the following
when SmartThings receives an event of contact closed" from
five different types of attacks.
the sensor. In addition, the state of an actuator's attribute is
Fake Events. They are events maliciously injected by at-
updated due to a feedback event, which is sent by the device
tackers. Fake events [80] may cause severe consequences by
after a command is executed by the actuator.
triggering actuator's actions. As illustrated in Figure 1(c), a
3
Motivation, Goals and Threat Model
fake presence-on event can unlock the door.
Fake Commands. An attacker may inject fake commands
IoT devices are notorious for their unreliability and insecu-
to IoT devices. For example, Sonos smart speaker [52] and
rity [25,40,46]. Numerous anomalies in appified homes have

WeMo Smart switch [62] accept commands from the local

been reported by users [4]. Below, we first discuss anomalies

network without authenticating their origins [58, 70].

due to IoT device malfunctions and attacks as the motivation,

and then present our goals and threat model.

Event Interceptions. Events can be intercepted and dis-

carded by attackers. E.g., the home security system can be

3.1

IoT Device Malfunctions

muted by intercepting the window and door sensors' wireless

connections to stop them from sending sensor events [66].

We survey real-world anomalies frequently reported in the

SmartThings user forum [4]. IoT devices interact with the IoT

Command Interceptions. Similar to event interceptions,

platform via events and commands; thus, we categorize mal-

an attacker can also intercept a command and prevents it

functions according to problematic events and commands.

from being delivered to the device [43].

Faulty Events. Faulty events refer to incorrect values re-

Compromised Devices. An attacker can compromise an

ported by IoT devices. They can be caused by sensor defects

IoT device and, at least, launch the following attacks. (1)

or physical interference, such as mysterious door-knocking

Stealthy Commands. The attacker can control the device

events [3] and motion events [9,17,46]. Faulty events may in-

to execute commands [65] and, to keep stealthy, stops the cor-

correctly trigger actuator actions and cause user confusions.

responding feedback events from being sent out. (2) Denial

of Executions (DoE). When a legitimate command is sent to

Ghost Commands. They are widely discussed in Smart-

the device, it does not execute the command but sends back

Things' user forum, dubbed 'poltergeists' [6,12,13]. For exam-

a feedback event reporting the command has been executed.

ple, a smart plug was turned on itself at night, which over-

heated the connected waffle maker and electrical grill [5].

3.3

Goals and Threat Model

Users frequently reported their lights were turned on during

midnight mysteriously [13].

We aim to detect both IoT device malfunctions described

in Section 3.1 and attacks in Section 3.2. We clarify that

Event Losses (or Large Delays). They refer to events that

HAWatcher can only detect attacks that violate correlations.

fail to be reported to the IoT cloud (in a timely manner). For

Attackers who have knowledge of the correlations may con-

example, mobile phone presence sensors were reported to

struct attacks that do not violate any correlations and thus

suffer from a large delay on status update [8], which was

evade our detection, which is discussed in Section 8.

confirmed by SmartThings [20]. Event losses may prevent the

execution of related automation and leave the home in risky

If feedback events are not muted, it is much like a Fake Command.