ESP: Using embedded sensors for Intrusion Detection

The evolution of host-based intrusion detection systems

Diego Zamboni
CERIAS, Purdue University

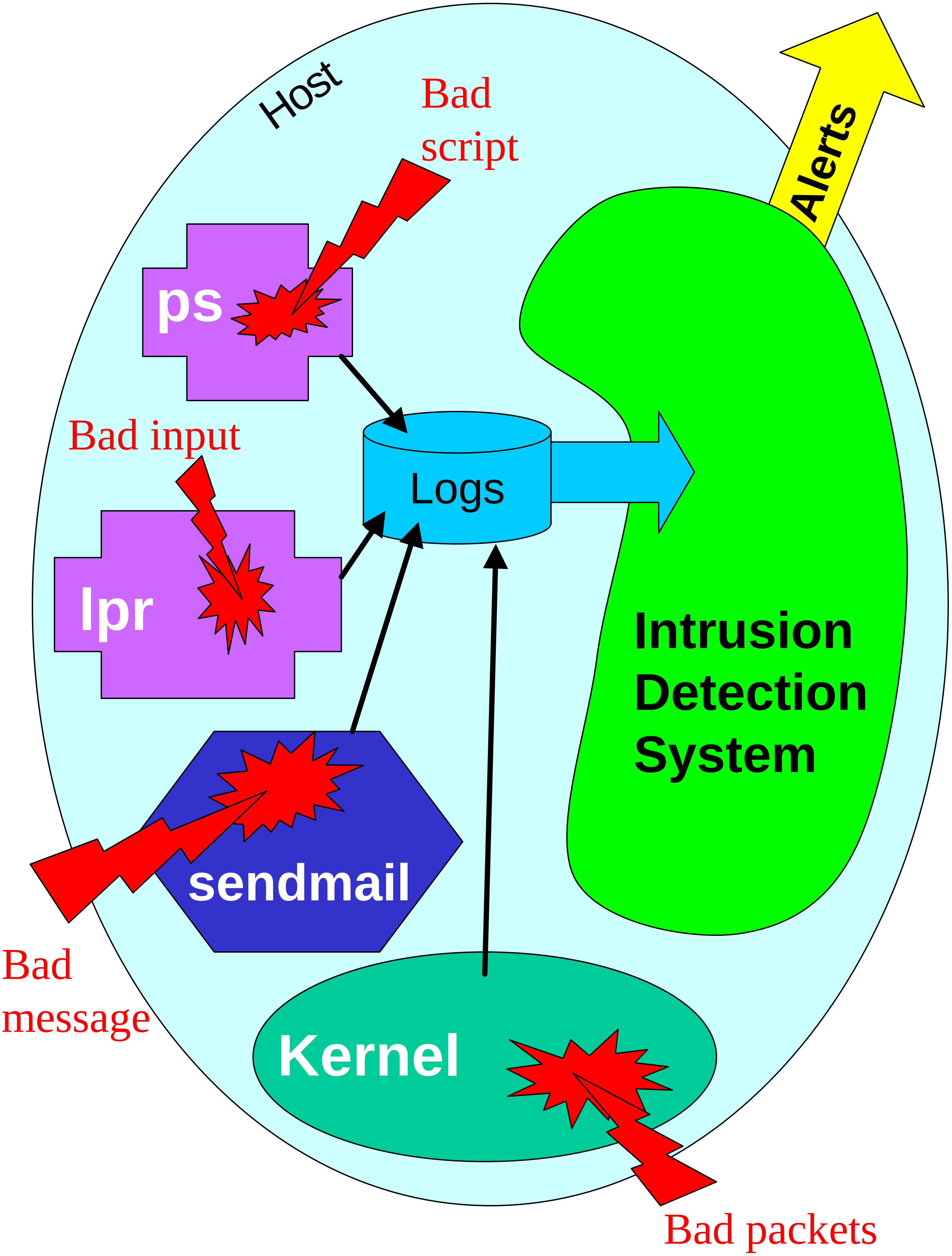
The middle ages: monolithic monitoring components

✓ Advantages:

- Can detect many problems
- Easy to set up (single component)

✗ Disadvantages:

- Big use of resources
- Continuous use of resources
- Cannot see everything
- Monitors through indirect means (audit trails)
- Single point of failure
- Hard to modify or add capabilities


Host
Bad script
ps
Bad input
lpr
Logs
sendmail
Bad message
Kernel
Bad packets
Intrusion Detection System
Alerts

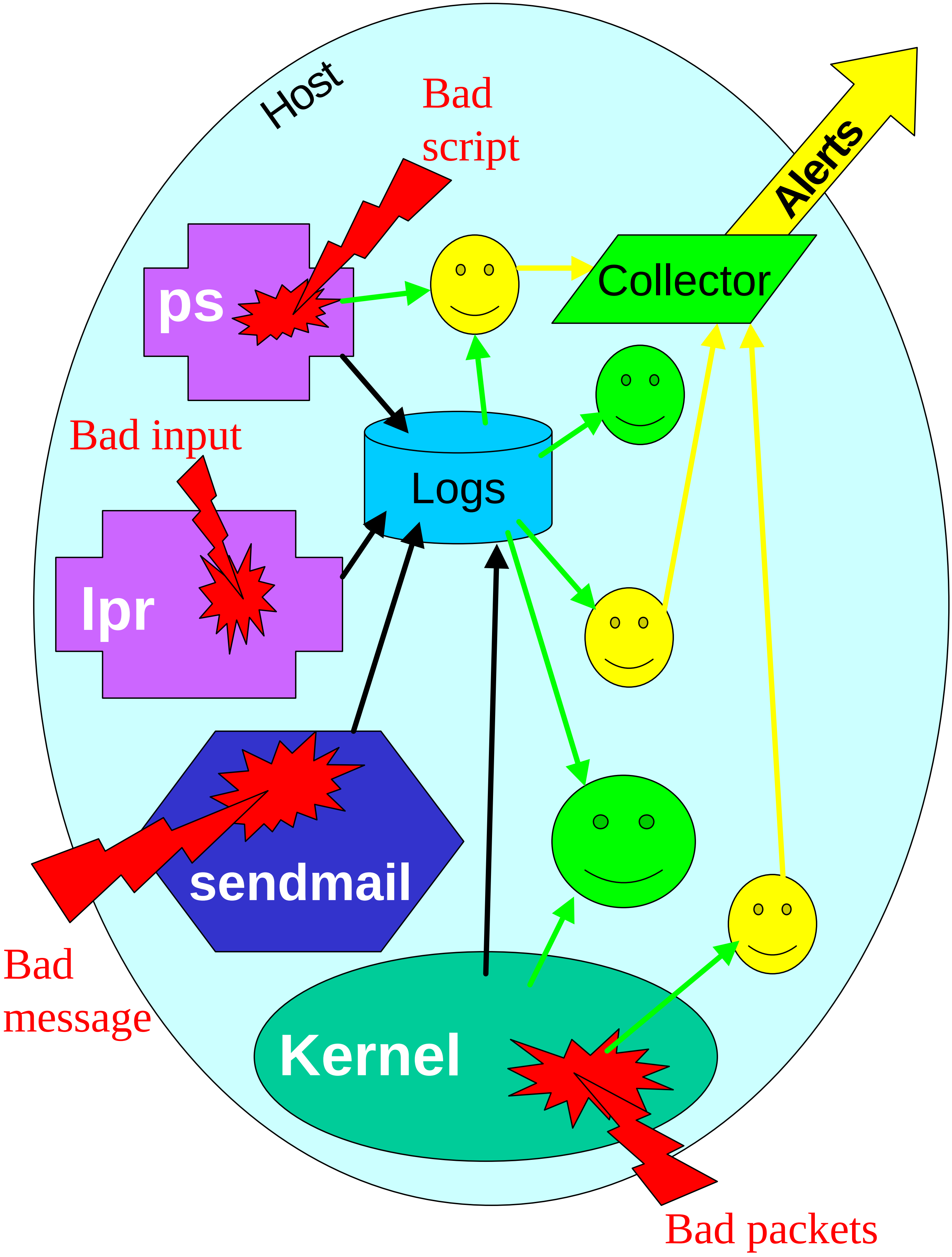
The renaissance: autonomous agents for monitoring

✓Advantages:

- Easier to add capabilities
- Lower resource usage
- Graceful degradation of service

✗Disadvantages:

- Complex to setup
- Difficult to correlate data
- Subject to tampering
- Monitor through indirect means (audit trails)
- Continuous use of resources
- Cannot see everything


Host
Bad
script
Alerts
ps
Collector
Bad input
Logs
lpr
sendmail
Bad
message
Kernel
Bad packets

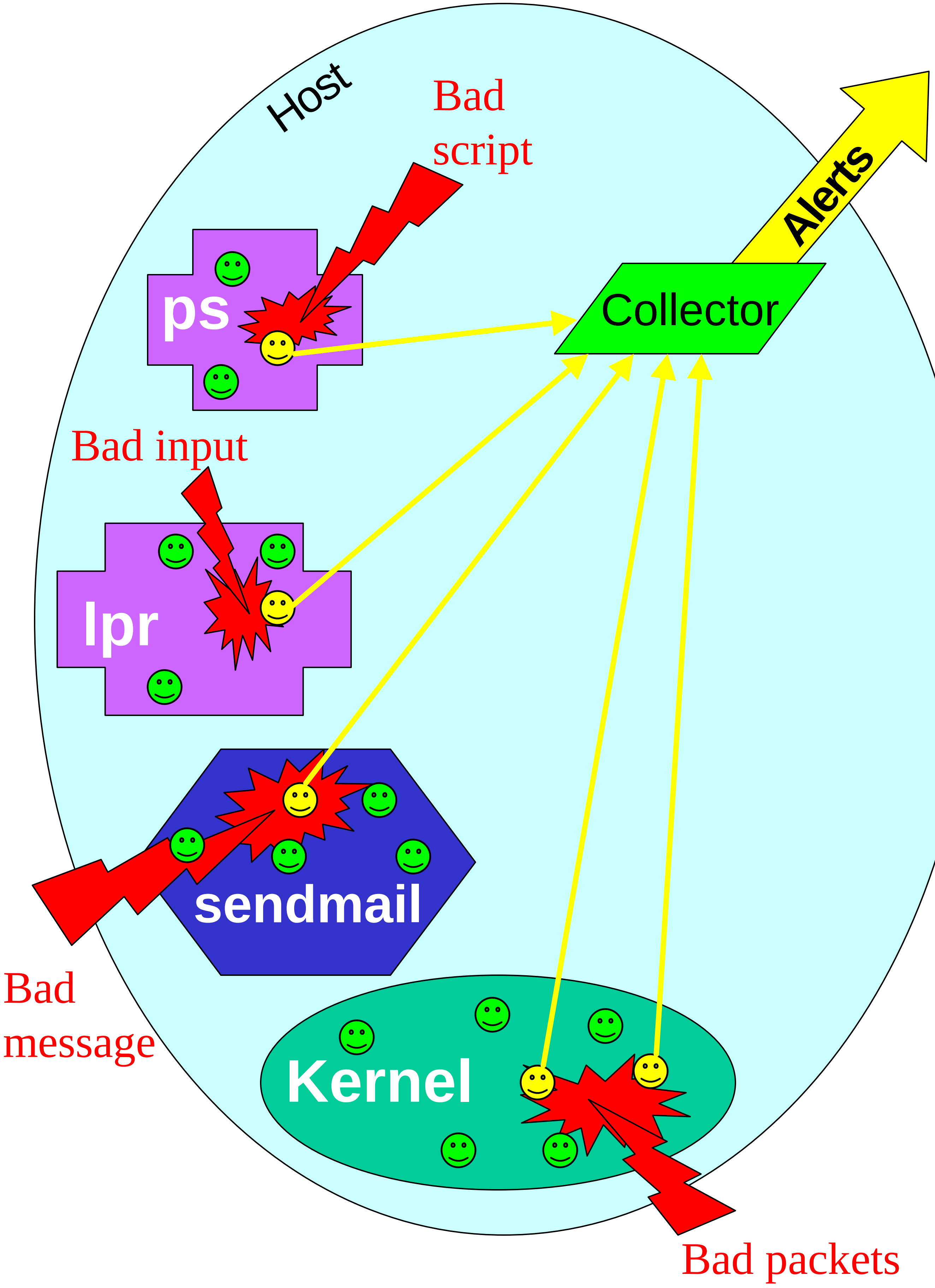
The new age: embedded sensors for monitoring

✓ Advantages:

- Almost zero extra resource usage
- Very difficult to tamper with
- Direct target monitoring (get data at the source)
- Can potentially see everything

✗ Disadvantages:

- Very system-dependent
- Need source code for the OS and its programs
- We do not know how to correlate data from different sensors


Host
Bad script
ps
Collector
Alerts
Bad input
lpr
sendmail
Bad message
Kernel
Bad packets

What is a sensor?

- A piece of code embedded in the affected program that looks for evidence of an intrusion

Before

```
char buf[256];
...
strcpy(buf,
 getenv(“HOME”));
...
```

After

```
char buf[256];
...
{ char tbuf[512];
  tbuf[0]=‘\0’;
  strncat(tbuf,
    getenv(“HOME”),511);
  if(strlen(tbuf)>255) {
   log_alert(“overflow”);
  }
}
strcpy(buf,
    getenv(“HOME”));
...
```

- We are using the CVE as a dictionary of vulnerabilities for which to build sensors
- Sensors are built into OpenBSD

What can we gain?

- Learn which types of data are more useful to detect intrusions, and where to collect them
- Learn how to build good sensors
- Stop depending on the data provided by the O.S. in its audit trails
- Learn if we can build a low-impact, highly reliable intrusion detection system
- See if we can detect new vulnerabilities with the existing sensors
- See if we can characterize intrusions