Fighting the Fog of War: Automated Incident Detection for Cloud Systems

Liqun Li, Xu Zhang, Xin Zhao, Hongyu Zhang, Yu Kang, Pu Zhao, Bo Qiao, Shilin He, Pochian Lee, Jeffrey Sun, Feng Gao, Li Yang, Qingwei Lin, Saravanakumar Rajmohan, Zhangwei Xu, and Dongmei Zhang

Microsoft Research, Microsoft Azure, Microsoft 365, The University of Newcastle, University of Chinese Academy of Sciences
Background

- Reliability is a key quality attribute of large-scale cloud systems
- Incidents/outages dramatically degrade the service quality
  - Tough incidents/outages take a long time to mitigate
  - Costs: $17K/outage·min (2016)*
Alerts are system events that require attention

- Reported by the monitoring infrastructure
  - E.g., API timeouts, operation warnings, unexpected VM reboots, etc.
- Severity: low, medium, high
- Handled by on-call engineers

![Graph showing the number of monitors and alerts from Big5 services in Azure]

Main fields of an alert

- Alert ID: 200603407
  - Title: Ongoing VM critical failures
  - Severity: High
- Service: Compute
  - Team: OS
  - Owner: Lily
- Start Time: 2020-03-13 07:30:00
  - Mitigation Time: 2020-03-13 08:23:00
- Region: A
  - Data Center/Cluster (Optional): xxx/xxx

Diagnosis

Logs

Monitor ID:

Data Center/Failure

*Big5: Compute, Storage, Networking, SQL DB, and Web Application*
Incident: situations with customer impact, taking a long time to resolve, or requiring cross-team collaboration

Timely incident management is the key to reduce system downtime

Incident declaration turns chaos into order

A long tail of incidents take a long time to declare

Engineers are like in the fog-of-war
Related Work

- Fault detection and localization
- Time-series anomaly detection
- Cloud incident management
Warden: Automated Incident Detection

- Detect the ongoing potential incidents from the alerts
- Extract incident-indicating alert groups for notification
Alert Signal Selection

- Select a subset of monitors which exhibit relatively strong association with incidents
  - Categorize incidents based on their responsible teams
  - Calculate the sum of Weighted Mutual Information (WMI) for each monitor with all subtypes of incidents

\[ I(s^m, i) \] is the information gain by observing alerts from a monitor \( m \) about predicting incidents of subtype \( i \)

The score of monitor \( m \) is \( \sum_i I(s^m, i) \)
Incident Detection

- **Incident detection: a binary classification problem**
  - Input: alerts reported by selected monitors in a recent time window
  - Output: 1 if there is potential ongoing incidents; otherwise, 0

- **Sample construction**
  - Construct samples using a sliding window (3h)
  - Label = 1 if the window is overlapping with incident impact duration; otherwise, label = 0

- **Feature extraction:**
  - Alert signals: alert count, alert burst
  - Engineer activities: diagnosis log count, notification count
  - Others: region, working day, hour in day

- **Classification model: BRF (Balanced Random Forest)**
Identifying the Incident-indicating Signals

- **Incident-indicating signals: alerts related to the incidents**
  - Alert signal grouping: correlation and rule-based
  - Group-based model interpretation: GSV (Group Shapely Value)

![Diagram showing incident-related alerts with impact start time and mitigation time](image-url)
Warden in Practice

- Detecting emerging issues
- Notify all engineers working on incident-indicating alerts
- Once confirmed, engineers form a cross-team collaboration group to diagnose and mitigate the incident
Experimentation

- 18-month-length ~240G data from Azure IcM
- 26 major services, Hundreds of people behind each service, ~72% of all incidents of Azure
- Training: 16 months; Testing: 2 months
- Baselines: Anomaly-H/S, AirAlert
**Experimentation**

- **Important system parameters**
  - # of selected monitors
  - Data requirements for training the detection model

![Graphs showing recall, precision, and F1 score vs. number of selected monitors, training data length, and interval](image-url)
Conclusions

- **Warden is a framework to detect incidents in an automated way**
  - Detecting potential incidents
  - Extracting related alert signals and notifying relevant engineers

- **Warden is proven to be effective with data collected from 26 major services and real deployment in the IcM of Azure**
Thanks