**Contingency plan for data corruption**

**1. Introduction**

Data corruption is a computer failure that occurs when an important document or a computer code stored on a support is modified by mistake. It can happen during:

- Writing,

- Reading,

- Storing (on a database or server) or

- Transmission of data

Data corruption can affect any type of support both internal and external.

Data corruption can be caused by an electronic fault, a mechanical problem or an electrical surge that prevents the storage support from having the information necessary to communicate with the host device. The data is then altered or damaged in an unpredictable proportion.

It is difficult to determine when a data corruption occurred or to quantify it, because it takes place inside the support or in the chips which control the communication between the support and the host device. Only its consequences on the data are visible: minor corruption will damage a handful of user files that will be lost or contain inaccurate information, while major corruption can affect the system area of a support at the risk of making it completely unusable later.

However, the possibility of significant or serious corruption remains very rare, especially with the most modern support which are equipped with measures to protect against data corruption and ensure their integrity from start to finish:

- Error correction code,

- Cyclic redundant check,

- Protection information standard.

**2. Preparation**

Security information and event management provides reports for unusual traffic and behaviors detected in network. They are grouped into 3 categories, based on threat level, in order to make them maintainable. Any incident or breach detected, classified at level 2 is saved separately. If incident or breach is well known and described, automated steps can be taken. If so, they are also reported.

Any incident or breach detected, classified at level 3 causes immediate alert to the first contact administrator. In this case, the first contact administrator checks, what resources have been compromised, and then takes further steps.

Other incidents are aggregated into one document, evaluated weekly. They are sorted into categories and increased grow in one of them may be the reason to initiate the investigation.

Any person involved in contingency plan, must be must be trained, in order to determine the method of communication. After serious incident, all people involved in the process are informed via company’s communicator, and text message. After that, an online meeting is taking place at company’s video communication platform, in specially prepared room. In case, when company’s network is unavailable, the meeting takes place directly at company’s building, in specified conference room. When one of people involved is unable to arrive within one hour, he informs about this fact the first contact administrator, and have to be on a standby on phone until the meeting is finished, in order to get information what steps needs to be taken.

**3. Detection and analysis**

The detection can be caused by hardware bugs which include bugs in the support or the disk shelf firmware, bad memory, and adapter failures. Software bugs could also cause some corruption. In many cases, the cause of corruption cannot be identified. We detect different forms of corruption using the different data protection mechanisms in place.

Here is a summary of some corruption classes:

- Checksum mismatches (CMs): This class refers to cases where the corruption is detected from mismatched data and checksum. The cause could be data content corrupted by components within the data path, or a torn write, wherein only a portion of the data block is written successfully, or a misdirected write, wherein the data is written to either the wrong support or the wrong location on the support, thus overwriting and corrupting data

Checksum mismatches can be detected anytime a support block is read (file system reads, data scrubs, RAID reconstruction and so on).

- Identity discrepancies (IDs): This class refers to a mismatch detected when a support block identity check is performed during a file system read.

The cause could be a lost write, which typically occurs because a write destined for disk is not written but thought of as written, or a misdirected write, where the original disk location is not updated.

We are aware of actual cases when the disk firmware replied successfully to a write that was never written to stable media. Identity discrepancies can be detected only during file system reads.

- Parity inconsistencies (PIs): This class refers to a mismatch between the parity computed from data blocks and the parity stored on the support despite the individual checksums being valid. This error could be caused by lost or misdirected writes, in-memory corruptions, processor miscalculations, and software bugs. Parity inconsistencies are detected only during data scrubs.

The detection of incident should be focused primarily on checksum mismatches, since we find that these corruptions occur much more frequently

We now describe our data collection and analysis methodology and some limitations.

**Data collection:**

The storage system has a built-in, low overhead mechanism to log important system events back to a central repository. These messages can be enabled for a variety of system events including disk errors. Not all customers enable logging, although a large percentage do. Those that do, sometimes do so only after some period of initial use. These logs allow customized support based on observed events. Although these logs are primarily intended for support, they have also been utilized for analyzing various disk errors. In addition to our corruption study, the repository used to store all logs has been used in disk failure and latent sector error studies.

**Analysis:**

We study corruption instances that were logged in tens of thousands of storage systems for a period of several months starting in January 2018. These systems belong to a range of different models, run different versions of storage-controller software (perhaps with one or more updates during the study period) and contain many different models or versions of hardware components. In order to have a complete history of the activities of the disks used in the study, we constrain our sample to only those supports that were shipped after January 2015. Our sample consists of 1.53 million support drives. These drives belong to 14 drives families and 31 distinct models. To derive statistically significant results, we often further constrain the sample set depending on the analysis being performed. For example, we sometimes use shorter time periods for our analysis so as to maximize the number of models we can study; clearly not all disk families and models have been in the field for the same duration. The disk models we consider for each study may have one of the following constraints:

- Model has at least 1000 disks in the field for time period being considered.

- Model has at least 1000 disks in the field and at least 15 corrupt disks for time being considered.

Since the sample size for different disk models per disk class varies considerably, we weigh the average by the sample size of each disk model in the respective class.

**Limitations:**

The study has a few limitations that mostly stem from the data collection process. First, for a variety of reasons, disks may be removed from the system. Our study includes those disks up to the point of their removal from the system. Therefore, we may not observe errors from otherwise error prone disks after some period of time. Second, since the logging infrastructure has been built with customized support as the primary purpose, the data can be used to answer most but not all questions that are interesting for a study such as ours. For example, while we can identify the exact disk when an error is detected during a scrub, we cannot verify that every disk in the study has been scrubbed periodically in the absence of errors.

Unavailability of any of the services offered by the system, when detected, is marked as maximum threat level. The first contact administrators are immediately informed about that fact, and manually take further steps. If it is possible ( implemented procedures includes such event), automated actions are taken ( like disconnecting device from company’s networks) in order to stop the spread of threat.

**4.Containment and alarming**

Appearance of incident or breath of third level of significance requires immediate action. First contact administrator is informed and starts the investigation. Having access to data connected with incident, he has to evaluate the seriousness and impact of problem.

After determining the sources of danger, and affected parties further actions are taken.

If alert turns out to be false (no action needed), then these information should be included in learning phase. Otherwise, the step against spreading of the issue are taken. Also, depending on initial threat analysis, the right people are put on standby.

1. If the corruption is detected, it has generally minor effect. A file may be lost, a bad hard drive sector may require repair, or a program may no longer function properly. Although troublesome, these problems are generally recoverable and limited in their scope. Even when they cause bigger problems, they at least give the user time to prepare. The most serious problems arise when corruption dissipates without being detected. In this case we can use the option of restoring from a backup or studying the problem further to see if the corruption was really random, or caused by a fixable problem.

2. If the corruption is undetected caused by some malware, hardware failures or buggy software; the problem must be taken seriously in the case that the source of the corruption was not addressed early in its life cycle. In a way, corruption is a cancer for computers. Detected early, it can be targeted and cut, but if it remains undetected it can multiply until suddenly the situation is hopeless

3. If problem concerns unavailability of services, the administrator has to check, if are they critical to the correct work of shop. If it is possible, and there is and existing procedure for such case,

admin can restart such service in order to make it work again. If it is not the case, such affected service should be isolated from the rest of the system during the investigation. Also, IT Manager is notified about that fact immediately. The website should be changed to previously prepared one, containing information about technical break and apology.

4. If problem concerns other system related threat, admin has to investigate the seriousness of it and affected assets. If unknown (not noted before) problem occurred, and possible damages are impossible to predict, IT Manager is notified immediately, and the meeting is taking place.

Generally, any incident reported, that can not be classified as false positive and cannot be handled automatically, leads to investigation. If basic services or data stored might be in danger, IT Manager is notified immediately. If there is a suspicion, than incident is intentional action of one of the employees, the Human Resources Manager is notified.

If possible sources of problem are found, the monitoring devices are tuned to recognize similar ones and report them for at least 7 days.

Next step is to identify damage caused. If data corruption is confirmed, then the recent backup should be used or checked in order to recover the file corrupted. In case of not successful recovering, the company can hired some external company specialized in data recovery and choose the premium plan proposed in order to get the recovered data in less time.

**5. Eradication and Recovery**

In cases, when IT Manager is involved his duty is to establish actions that will allow system recovery. Provided information from first contact administrators, he has to found out the source of problem, identify the consequences and contact the Crisis Manager in order to get authorization for

actions needed. With help of his team, he performs:

* neutralize the danger: depending on situation in may be blocking specific requests, that led to unauthorized actions
* in case of service loss: restarts affected services, recovers system back to work
* after data corruption: checking backups in order to restore data, estimates the size of data that cannot be restored

All estimations are forwarded to the Crisis Manager, who is responsible to take eventual business, legal or other actions, supported by Critical Operations Support Staff.

In case, when the company can’t successfully recover the data, some calls to external organizations specialized in data recovery should be placed.

After all, IT Manager creates report that will allow to avoid such incidents in the future. This may include changes in procedures, including new ones, that will allow to automate similar incidents in the future, or new security features that should be implemented in the system in order to prevent.

In cases where the IT Manager is not called, the administrator is responsible for fixing the problem.

If incident, that is being monitored starts to appear more frequently, First Contact Administration is obliged to inform the IT Manager about this fact.

**6. Learning**

Learning phase includes usage of information acquired during the alert. In general, nearly half of alerts are so called false positives, that do not require action, but only wastes the time of First Contact Administrators who have to check them.

After alert, that found out false, there should be applied tuning to the detection system. It may include the higher rate of similar incidents needed to trigger the alert – connected for example with growing number of users. Finding correlated assets and binding them together is one of the ways.

The tuning should be done continuously.

The recommendations in reports that are created by IT Manager after recovery should be included in the system as soon as possible. The Recovery Management Team is subject responsible for applying changes, after they are confirmed by Crisis Manager.