Summary

Zabbix 5.2 introduced new trend functions useful for baseline monitoring. However, they still require defining relative thresholds (e.g. check that web traffic in September, 2021 is less than 2x higher compared to September, 2020). There are use cases when such thresholds are hard to define. For instance, the web traffic of a new but highly popular web site can organicaly grow many times over a year but the growth rate is unknown. Yet, a sudden traffic spike due to DDOS attack must generate an alert regardless of organic traffic growth.

Anomaly detection algorithms do exactly this - find data that don't look normal (outliers) in a context of other values. This AC specifies anomaly detection by the method called decomposition. From the start Zabbix must support one decomposition algorithm - STL. More anomaly detection algorithms will be added in the future.

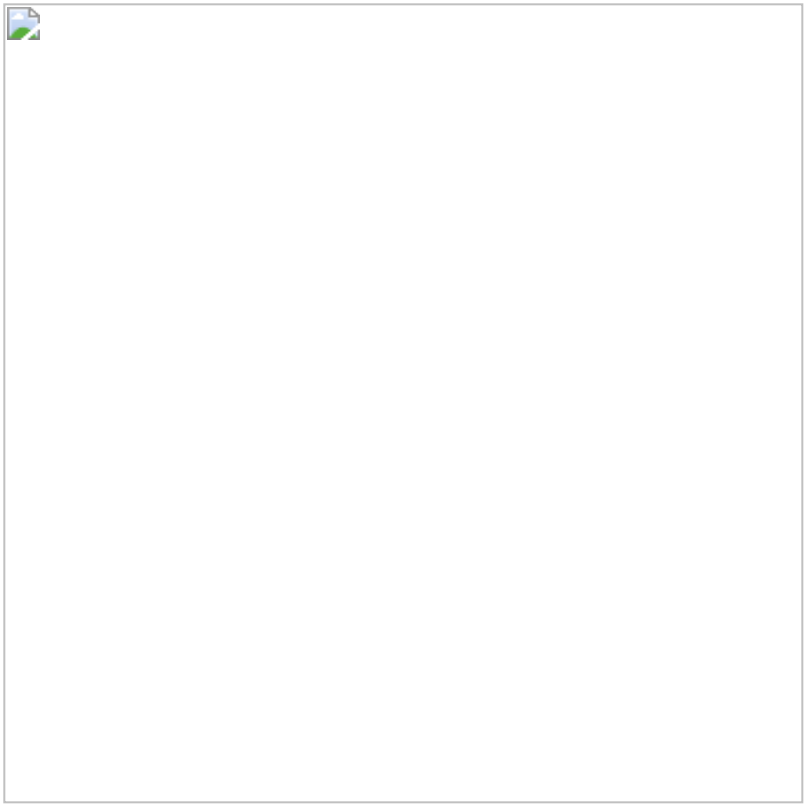
Use cases

1. I want to detect anomalies in my data.

General idea

Decomposition is a way to split a single time series sequence into three other sequences:

- trend sequence that only contains big changes in the original data (e.g. website traffic shows growth)
- season sequence that only contains seasonal changes (e.g. less website traffic in summer, more in autumn)
- remainder sequence that only contains residual values that can not be interpreted as parts of trend or season



Anomaly detection works with remainder sequence and checks if there are values that are too far from the majority of remainder values. "Far" means that the absolute value from the remainder sequence is N times greater (see *deviations*parameter below) than the standard or mean deviation.

Obviously, the more values participate in the standard/mean deviation calculation, the better is anomaly detection. But, typically, we are interested in more recent values, there is no point in reporting anomalies in the old data. That's why we need two periods to calculate and report anomalies: *eval period* provides the time segment for decomposition and *detect period* provides the segment to report.

Zabbix acceptance

1. Zabbix must support anomaly detection by implementing decomposition algorithm STL
    a. New trend functions for triggers and calculated items:
        i. trendanomalystl(/host/key, eval period:<time shift>, <deviations>, <devalg>, <detect period>, <season>, <parameters>)
            1. Works only with trend data (value_avg)
            2. Returns a decimal value between 0 and 1 that is *number of anomaly values / number of values in detect period* (anomaly rate)
            3. Parameters:
                a. eval period:<time shift> - time period that must be decomposed
                b. deviations - the number of deviations (calculated with devalg) to count as anomaly; can be decimal
                c. devalg - deviation algorithm; 'stddevpop', 'stddevsamp', 'mad' (default)
                    i. Algorithms are the same as stddevpop(), stddevsamp(), mad() trigger functions
                d. detect period - time period at the end of eval period
                    i. Anomalies calculated for this period
                e. season - length of a season ($n_{(p)}$ - per original description) expressed as time period
                    i. season is a shortest period where seasonality (repeating patterns) is expected
                f. parameters - additional list of STL-specific parameters (such as $n_{(s)}$, $n_{(t)}$, etc)
                    i. Default values must be proposed (see these explanations and 6.1 of original description)
            4. The way to handle missing data must be proposed in the specification
        ii. trendanomaly(/host/key, eval period:<time shift>, detect period, season)
            1. Alias for trendanomalystl(/host/key, eval period:<time shift>, 3, "mad", detect period, season)
    b. Front-end must be modified to support the new functions

Example:

- `trendanomalystl(/Zabbix server/system.cpu.load, 30d, 4, "mad", 1d, 1d)`
    - Calculate anomaly rate for the last day by decomposing trend data from the last 30 days (by using STL)
        - anomaly values are greater than 4 * mean absolute deviations (mad)
        - mad is calculated for the remainder sequence of last 720 values (24 * 30)
        - use seasonality 1 day
        - if mad = 0.1 and there are 2 values greater than 0.4 among the last 24 (because 1d is 24 values in trends) then trendanomalystl() = 0.083333

Zabbix UI changes

1. N/A

Decisions made

1. Do not provide fine-grained control over decomposition (no separate decompose(), mad() etc) because we do not want to introduce the notion of time series vectors in the trigger functions.
2. These functions will work with trends only.
3. No time shift for detect period.
4. No template modifications at this time.

Open questions

1. N/A

Changes log

- N/A