The timestamped results are found by looking at stdout from the "vul_prog" program, and finding the first line that starts with: "The original secrets:". The result is assigned the value of the sixth space-delimited token in that line. The "newsecret1value" assignment is similar. The goals.config file includes:

```
modify_value = matchany : string_diff : newsecret1value : result.origsecret1value
```

, which will be TRUE if any of the vul_prog stdout files include a "newsecret1value" that differs from its "oldsecret1value".

6.6.4 Was a log entry written while some command executed?

Consider these two entries in `results.config`:

```
# Time stamp of log entry containing IP address
log-from-w1 = w3:/var/log/myhttplogfile.txt : LOG_TS : 202.25.4.2
# Use of wget -- will result in time stamp range: start-finish
wget-w1 = w1:wget.stdin : CONTAINS : 202.25.4.2
```

The following `goals.config` entry will be true if the log entry was ever generated using wget from the w1 computer:

```
didit = time_during : log-from-w1 : wget-w1
```

6.6.5 My desired artifacts are not in stdin or stdout, the program outputs a file

See section 6.1.2

6.6.6 Delimiting time using log file entries

The LOG_RANGE result type generates a set of results having timestamp ranges that cover the period between specified log entries. For example, a results.config directive of:

```
syslog_slices = server:/var/log/messages : \
    LOG_RANGE : Started System Logging Service
```

would create a set of time ranges with periods between each start of the system logging service. The use of `time_during` and/or `time_not_during` and `boolean` in the goals.config could then assess whether two or more events occurred during a given system log configuration. For example, assume the results.config file also included these directives:

```
_did_first_thing = client1:did_this.stdout : CONTAINS : Did that thing
_did_second_thing = client2:did_other.stdout : CONTAINS : Did that other thing
```

We'd like to know if the above two results were ever achieved within one configuration of the logging system. This can be determined by first binning the above two results into the time ranges established by the `syslog_slices` result through use of `time_during` within the goals.config as follows.

```
_did_first_during = time_during : _did_first_thing : syslog_slices
_did_second_during = time_during : _did_second_thing : syslog_slices
```

That yields two sets of goals having time ranges defined by the `LOG_RANGE` results. We can then use a boolean operator to determine if those two goals were ever achived within the same established time range[21]:

```
did_both = boolean : (_did_first_during and _did_second_during)
```

See the centos-log2 lab for an example.

[21] Recall that the use of the boolean operator only makes sense for goals/results having matching timestamps