MONITOREO DEL ACTIVE DIRECTORY CON KIBANA-ELASTISEARCH

Descripción de la arquitectura

Para realizar el monitoreo, se debe instalar “Winlogbeat” en el mismo servidor donde está el Active Directory.

WinlogBeat envía los eventos del Active Directory al ElasticSearch (Data-Lake), para acceder a los logs en el ElasticSearch se utilizara el Kibana, que es viene a ser una página web.

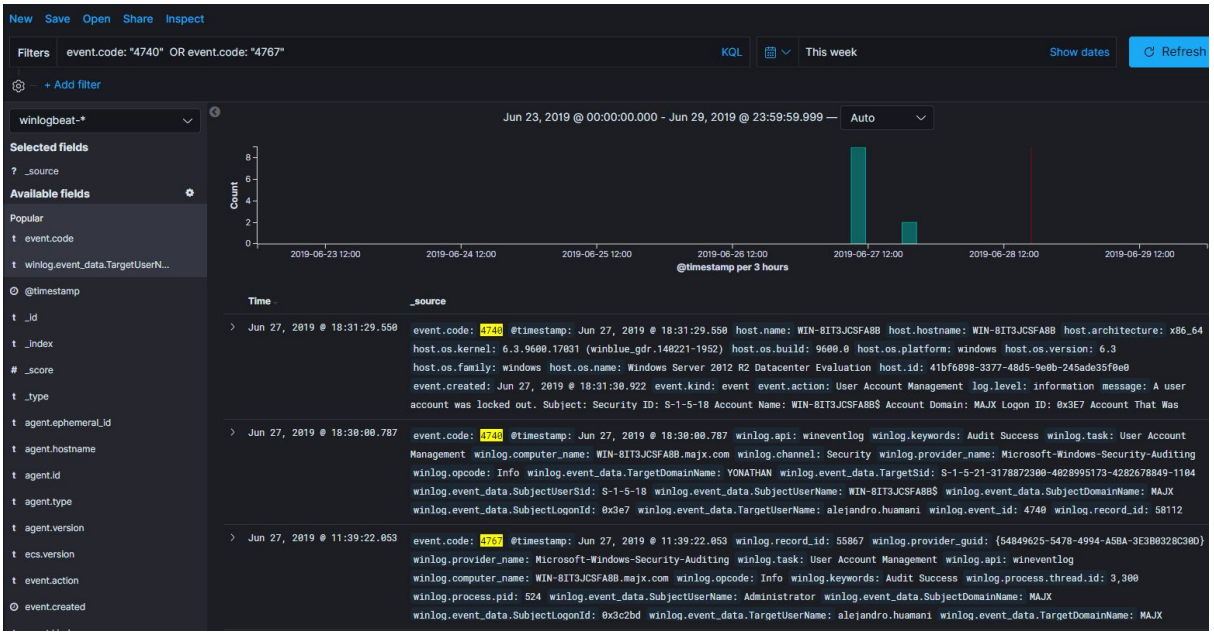
Visualización de eventos

Los eventos que son generados por el “Active Directory” son almacenados en un data-lake, denominado ElasticSearch, para navegar a través de estos datos se utiliza el navegador web, cuyo fronted es “Kibana”.

Visualización de Logs en el “Discovery”

El Discovery es una de las herramientas del Kibana que da acceso a los logs (datos en crudo generados por Active Directory) El Discovery, tiene una interfaz intuitiva, para facilitar el filtrado de los datos, según se requiera.

A continuación se muestra los logs recibidos, en un rango de una semana y filtrado por el código del evento: 4740 (Bloque) y 4767(Desbloqueo).

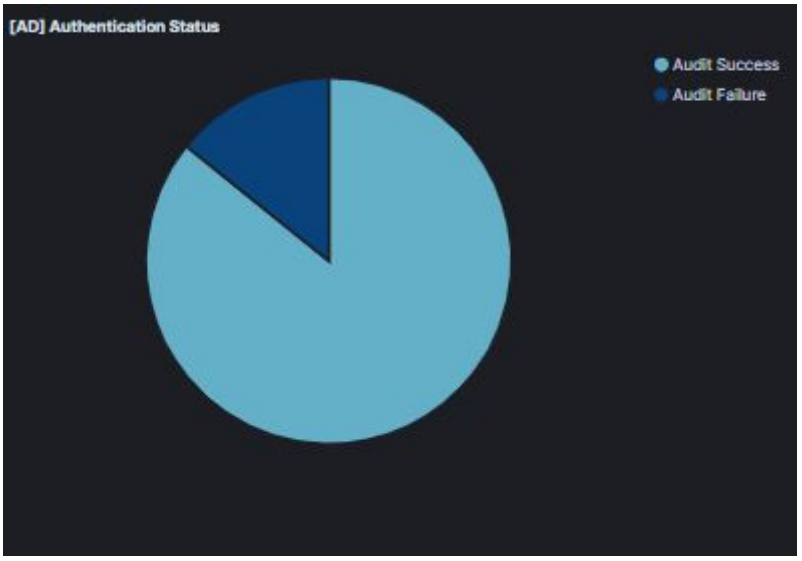


Visualización de Indicadores de interés

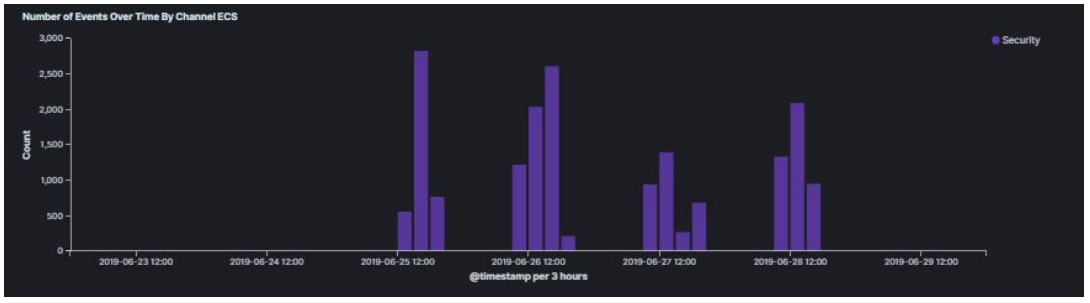
Además del Discovery, el Kibana permite crear Dashboards a medida del cliente, a continuación se muestra un resumen del estado de las cuentas en el “Active Directory”.

1. Resumen de la cantidad de cuentas bloqueadas y no bloqueadas.

2. Porcentaje de acceso fallidos y exitosos

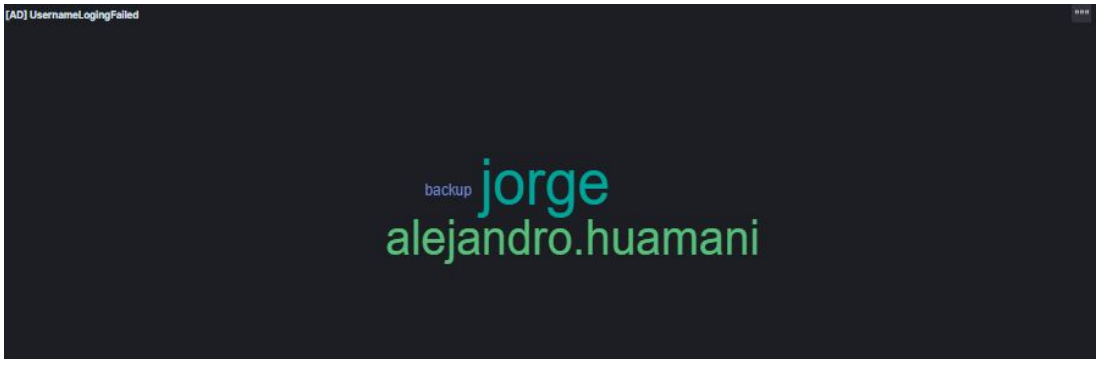


3. Eventos de seguridad generados por el Active Directory en una semana.



4. Top de cuentas bloqueadas por un rango mayor a 1 día (este valor puede ajustarse).

5. Top de username con más errores de login.

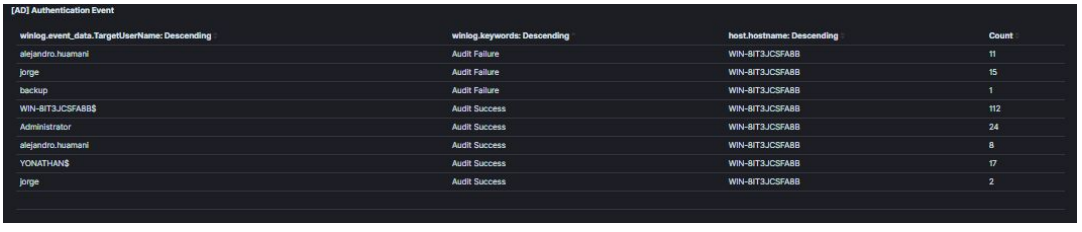


6. Tabla resumen del estado actual de las cuentas gestionadas por el Active Directory


[AD] Status Account

_id: Descending	status: Descending	Count
jorge	bloqueada	1
alejandro.huamani	desbloqueada	1
javier.huamani	desbloqueada	1

7. Tabla de los username clasificados por acceso (“failure” , “Success”).


[AD] Authentication Event

winlog.event_data.TargetUserName: Descending	winlog.keywords: Descending	host.hostname: Descending	Count
alejandro.huamani	Audit Failure	WIN-8IT3JCSFA8B	11
jorge	Audit Failure	WIN-8IT3JCSFA8B	15
backup	Audit Failure	WIN-8IT3JCSFA8B	1
WIN-8IT3JCSFABB$	Audit Success	WIN-8IT3JCSFA8B	112
Administrator	Audit Success	WIN-8IT3JCSFA8B	24
alejandro.huamani	Audit Success	WIN-8IT3JCSFA8B	8
YONATHAN$	Audit Success	WIN-8IT3JCSFA8B	17
jorge	Audit Success	WIN-8IT3JCSFA8B	2

Detalle del watcher

```
POST _watcher/watch/ad_blocked_account
{
    "trigger" : {"schedule" : {"interval" : "1m"}},
    "input" : {
      "chain" : {
        "inputs" : [
          {
            "chain_01" : {
              "search" : {
                "request" : {
                  "search_type" : "query_then_fetch",
                  "indices" : ["winlogbeat-7.2.0"],
                  "rest_total_hits_as_int" : true,
                  "body" : {
                    "size" : 0,"sort" : [{"@timestamp" : {"order" : "desc"}}],
                    "_source" : ["_id", "event.code", "winlog.event_data", "@timestamp"],
                    "query" : {
                      "bool" : {
                        "should" : [
                          {"match" : {"winlog.event_id" : 4740}},
                          {"match" : {"winlog.event_id" : 4767}}
                        ]
                      }
                    },
                    "aggs" : {
                      "groupbytype" : {
                        "terms" : {"field" : "winlog.event_data.TargetUserName", "size" : 10},
                        "aggs" : {
                          "event_type" : {
                            "terms" : { "field" : "winlog.event_id","size" : 10 },
                            "aggs" : {
                              "SOURCE" : {
                                "top_hits" : {
                                  "_source" : [ "event","winlog","@timestamp"],"size" : 1,
                                  "sort" : [{"@timestamp" : {"order" : "desc"}}]
                                }
                              }
                            }
                          }
                        }
                      }
                    }
                  }
                }
              }
            }
          }
        ]
      }
    },
    "condition" : {
      "script" : {
        "source" : """
  def numero_secuencial = ctx.payload.chain_01.hits.total;
  def messages = [];
```

```
  for (def account: ctx.payload.chain_01.aggregations.groupbytype.buckets) {
    def list_event_id = account.event_type.buckets;
    def event_data = list_event_id[0].SOURCE.hits.hits[0]._source.winlog.event_data;
    def user_name = event_data['TargetUserName'];
    def domain_name = event_data['TargetDomainName'];
    def latest_blocked_date = null;
    def latest_unblocked_date = null;
    def latest_blocked = "2019-01-01T00:39:22.053Z";//"
    def latest_unblocked = "2019-01-01T00:39:22.053Z";//"
    def sdf = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss.SSS'Z'");// "
    def flagblocked = false;
    def flagunblocked = false;
    def status_account = "desbloqueada";
    def doc_id = 0;
    for(def one_event: list_event_id){
        def event_id = one_event.SOURCE.hits.hits[0]['_source']['event']['code'];//one_event.key;
        def timestamp = one_event.SOURCE.hits.hits[0]['_source']['@timestamp'];
        //Cuenta bloqueada
        if(event_id == 4740){
          latest_blocked = timestamp;
          doc_id = one_event.SOURCE.hits.hits[0]['_id'];
          flagblocked = true;
        }
        //Cuenta Desbloqueada
        if(event_id == 4767){
          latest_unblocked = timestamp;
          flagunblocked = true;
        }
    }
    latest_blocked_date = sdf.parse(latest_blocked).getTime();
    latest_unblocked_date = sdf.parse(latest_unblocked).getTime();
    //if(latest_blocked_date>latest_unblocked_date){
    if (flagunblocked){
      if(latest_blocked_date>latest_unblocked_date){
        //La cuenta aun no se desbloquea
        status_account = "bloqueada"
      }
    }
    messages.add([
      "_id": user_name,
      "user_name": user_name,
      "domain_name": domain_name,
      "latest_blocked": latest_blocked.toString(),
      "latest_unblocked": latest_unblocked.toString(),
      "flagblocked": flagblocked,
      "flagunblocked": flagunblocked,
      "status": status_account
    ]);
  }
    ctx.payload._doc = messages;
    return messages.size()>0;
""",
        "lang" : "painless"
      }
    },
    "throttle_period_in_millis" : 1000,
    "actions" : {"index_payload" : { "index" : {"index" : "active_directory", "doc_type" : "_doc"}}},
    "metadata" : { "window" : "2m"}
  }
```