# Robust “Impossible Travel” in Splunk — Risk-Based Enrichment (SPL Included)

Two complementary approaches to high-fidelity detections: **triage + enrich** Defender alerts, and a **direct detector** on Entra sign-ins with adjacent pairing, haversine distance/speed, and intel/VPN risk fusion.

SPL Included

PREREQUISITES

- Lookup: `abuseipdb_all_ips` (fields: `ip`, `abuseConfidenceScore`)
- Custom command: `ipqs` (emits `ipqs.vpn_and_proxy_score`)
- Adjust `index`/`sourcetype` and field names to your telemetry; `coalesce()` guards keep queries resilient.

## SPL A — Triage Defender “Impossible travel” Alerts (intel + VPN risk)

```
/* Purpose: Triage Defender "Impossible travel activity" alerts with AbuseIPDB + IPQualitySc
   Pre-reqs:
     - Lookup: abuseipdb_all_ips (fields: ip, abuseConfidenceScore)
     - Custom command: ipqs (emits ipqs.vpn_and_proxy_score)
   Notes: Field names vary by source; use coalesce() defensively. */

index=azure sourcetype=azure:defender:cloudalert alert_type="Impossible travel activity"
| eval user=coalesce(user_principal_name, user, tostring(properties.userPrincipalName,""))
| eval dest_ip=coalesce(properties."destination.ip", tostring(additionalFields.DestinationIP
| where isnotnull(user) AND isnotnull(dest_ip)
| lookup abuseipdb_all_ips ip AS dest_ip OUTPUTNEW abuseConfidenceScore
| ipqs ip=dest_ip
| rename ipqs.vpn_and_proxy_score AS vpn_score
| eval is_vpn=case(isnull(vpn_score),"Unknown", vpn_score>=75,"Yes", true(),"No")
/* Simple triage score: Defender baseline + intel + VPN */
| eval intel_score=coalesce(abuseConfidenceScore,0)
| eval triage_score=round(20 + intel_score + if(is_vpn="Yes",25,0),0)
/* Keep only materially risky alerts */
| where triage_score>=50
```

```
| table _time user dest_ip abuseConfidenceScore vpn_score is_vpn triage_score alert_severity
| sort - _time
```

## SPL B — Direct “Impossible Travel” from Sign-ins (adjacent pairing + haversine + risk)

```
/* Purpose: Detect adjacent user sign-ins that imply impossible travel; enrich with AbuseIPD
   Works with Entra ID / Azure AD sign-in telemetry. Adjust index/sourcetype to your environ
   Thresholds:
     - distance_km >= 500 AND speed_kmph >= 900 → "impossible"
   Pre-reqs:
     - Lookup: abuseipdb_all_ips (ip → abuseConfidenceScore)
     - Custom command: ipqs (→ ipqs.vpn_and_proxy_score) */

(
  index=azure sourcetype=azure:signinlogs ResultType=0
  OR index=o365 sourcetype=o365:management:activity Workload=AzureActiveDirectory Operation=
)
| eval user=coalesce(user, user_principal_name, userPrincipalName, ActorUPN)
| eval src_ip=coalesce(src_ip, client_ip, ipAddress, ClientIP, SourceIPAddress)
| where isnotnull(user) AND isnotnull(src_ip)
/* Geolocate current login */
| iplocation prefix=src_ src_ip
| where isnotnull(src_lat) AND isnotnull(src_lon)
/* Pair with immediately previous login for the same user */
| sort 0 user _time
| streamstats current=f window=1 last(_time) AS prev_time last(src_ip) AS prev_ip last(src_l
| where isnotnull(prev_time) AND src_ip!=prev_ip
| eval delta_min=round((_time - prev_time)/60,1)
| where delta_min>0 AND delta_min<=1440
/* Haversine distance (km) */
| eval dlat=((src_lat - prev_lat)*pi()/180), dlon=((src_lon - prev_lon)*pi()/180)
| eval a=pow(sin(dlat/2),2) + cos(prev_lat*pi()/180)*cos(src_lat*pi()/180)*pow(sin(dlon/2),2
| eval c=2*asin(min(1, sqrt(a)))
| eval distance_km=round(6371*c,1)
/* Speed (km/h) and impossibility */
| eval speed_kmph=round(distance_km/(delta_min/60),1)
| eval impossible=if(speed_kmph>=900 AND distance_km>=500,1,0)
/* Intel enrichment on current src_ip */
| lookup abuseipdb_all_ips ip AS src_ip OUTPUTNEW abuseConfidenceScore AS abuse_score
| ipqs ip=src_ip
| rename ipqs.vpn_and_proxy_score AS vpn_score
```

```
| eval is_vpn=case(isnull(vpn_score),"Unknown", vpn_score>=75,"Yes", true(),"No")
/* Risk fusion: distance/speed + VPN + intel */
| eval risk = (impossible*60) + (case(vpn_score>=75,20,true(),0)) + round(coalesce(abuse_sco
| where impossible=1 OR risk>=60
| table _time user prev_time prev_ip src_ip prev_country src_Country delta_min distance_km s
| rename src_Country AS country
| sort - _time
```

NOTES

- Tune `vpn_score` and `abuseConfidenceScore` thresholds to your baseline; raise/lower triage/risk cutoffs as needed.
- If you run Splunk ES with RBA, append: `| eval risk_object=user, risk_score=coalesce(triage_score, risk)` and save as a correlation search.
- Prefer adjacent-login pairing over earliest/latest to avoid false pairs.

EXPORT TIP: `CTRL/CMD` + `P` → SAVE AS PDF → UPLOAD TO LINKEDIN AS A **DOCUMENT**.