--The Issue --

In networks with sufficiently complex p2p networks, full 20mb blocks and a full mempool yield a condition where chains stop producing blocks reliably. Further, initiating this condition results in a secondary impact, whereby the nodes in the victim network continue to try to produce blocks, forcing each other into states where proposals fail chain-wide after 336 block parts accrue.

This can be divided into a 4 different scope categories, shown below.

Cometbft team:
-consensus issues resulting in an invalid proposer signature error
-p2p issues of an unknown type, could be a lack of throttling or ALG controls

Ibc-go team:
-client spamming
-lack of field length limits in ibc memos
-lack of field length limits in ibc for receiver in send transactions
-banana king -reported 1 year prior, published on Twitter 6 months prior- and is referenced in this report

Cosmos-sdk team:
-gas fees above 0 do not limit the attack significantly
-mempool echoes / amplification seen 30+ minutes after attack conducted, including next day
-invalid block parts error at 336 block parts, amplifying stuck state to peers

[not for ownership of these groups] scopes for chain teams and/or individual operators:
-ability to dos through globalfee whitelists, even with a 1mb block size
-maximum block size

Especially vulnerable

-Juno
https://github.com/CosmosContracts/juno/blob/b03500a3f4d7cea9d6d8de11e0a63c6d92502dc2/app/app.go#L491
-Cosmos Hub
https://github.com/cosmos/gaia/blob/a4eb27a3831340c9505653012c8e3f204a90961d/x/globalfee/types/params.go#L24-L33
-Noble
https://github.com/strangelove-ventures/noble/blob/fcb702cda3c8fcc7982a25ac78eb3723edd45f19/x/globalfee/types/params.go#L22

--Vulnerable

Every cosmos chain is vulnerable to this, provided that the attacker is willing to pay for downtime.

Killchain already utilized in POC in production 2+ times:

[ROBIN: I have moved this information to the central P2P storms doc and written it up as]
1. Attacker controls a moderate amount of capital
2. Attacker wants to make money on a 30x short
3. Attacker floods the hub with invalid client update transactions or similar
4. Attacker gets the hub to make 80 second blocks ie Stride (2023)
5. Attacker and co-conspirators flood x.com with the fact that this is happening
6. Users fud and inflame the chain during the incident with more tx attempts
7. incident continues until a new majority of validators update to limit block size and resync states as the chain struggles with p2p peer stability

Stride - Aug 14, 2023 to Aug 19, 2023
Observed symptoms:

-High block time
-High rate of proposal failure
-Nodes unable to get the whole block
-Blocks filled with invalid client updates
-Full blocks containing 336 block parts
-Full Mempool
-Block size of about 20 megabytes
-Bandwidth consumption of 500mbps in both directions
-Range security mentions that blocks were full of invalid client updates

Impacts and Records of Attack Success:

Regarding questions relating to gas for client update transactions:
https://github.com/cosmos/gaia/blob/a4eb27a3831340c9505653012c8e3f204a90961d/x/globalfee/types/params.go#L24-L33

Easiest means of reproduction

1. Find tendermint networks with sufficiently complex p2p networks (including Cosmos Hub)
2. Spam until the mempool is full of either valid or invalid transactions, exploiting the unrestricted blocksize at 20-22mb

Completed successfully 2 or more times on Cosmos testnet 26th September 2023; some validators didn't recover many hours later. Completed multiple times leading up to Cosmoverse to attempt to get alternate responses from security researchers and other teams.
Example attack results after the attack execution has ended for more than 15 minutes:
https://ibb.co/Fwqk0wY

No mainnet attack has been conducted by the reporter, however, as whitehats we are compelled to attack the Cosmos Hub testnet for validation of this vulnerability.

NOTE: the methods listed below are likely non-exhaustive. For example, Terra Classic was flooded with contract transactions. The critical factors to perform this attack successfully are:
-large (20mb) blocks and a
-full mempool
-peer to peer network active accross multiple regions

Seen in current repos:
Juno
https://github.com/CosmosContracts/juno/blob/b03500a3f4d7cea9d6d8de11e0a63c6d92502dc2/app/app.go#L491
Cosmos Hub
https://github.com/cosmos/gaia/blob/a4eb27a3831340c9505653012c8e3f204a90961d/x/globalfee/types/params.go#L24-L33
Noble
https://github.com/strangelove-ventures/noble/blob/fcb702cda3c8fcc7982a25ac78eb3723edd45f19/x/globalfee/types/params.go#L22

Validated methods of reproduction, actively tested on Cosmos hub testnet 26th September 2023, and multiple times before the end of the month.

-Banana King
Oversized IBC transactions that are valid due to a lack of field length limitations in ibc, failure of constraints in cosmos-sdk and failure of maxtxbytes in comet. This was observed March 14th 2023 on Osmosis.
-Client update spam
Spam client updates until blocks are full, as seen on Sentinel (Q4 2021) and Stride (2023)
-Use the lack of field length limitations in ibc to create client updates that are 1-20MB in size each

The banana client attack should have no cost on Cosmos Hub, Noble or Juno

Banana attack cited on Twitter 14 March 2023 with references to prod tx:

https://web.archive.org/web/20230926135340/http://web.archive.org/screenshot/https://twitter.com/web3_analyst/status/1635687287962112000

https://web.archive.org/web/20230926135641/http://web.archive.org/screenshot/https://twitter.com/web3_analyst/status/1635687292345147393 (link to explorer: https://www.mintscan.io/osmosis/tx/D62F0F0354C4DEA0D9DFCA596D9BC3F2943DBA7D24818009DFD725F883088DD0)

https://web.archive.org/web/20230926135831/http://web.archive.org/screenshot/https://twitter.com/web3_analyst/status/1635687294991732736

https://web.archive.org/web/20230926135831/http://web.archive.org/screenshot/https://twitter.com/web3_analyst/status/1635687297608994817

Resources demonstrate the impact of the attack on testnet:

https://web.archive.org/web/20230926135439/http://web.archive.org/screenshot/https://monitor.polypore.xyz/d/rYdddlPWk/node-exporter-full?orgId=2&var-DS_PROMETHEUS=default&var-job=provider-1&var-node=provider-apple.rs-testnet.polypore.xyz:9100&var-diskdevices=%5Ba-z%5D%2B%7Cnvme%5B0-9%5D%2Bn%5B0-9%5D%2B

Brief writeup on Sentinel 2021 incident, HackerOne report available upon authorized request: https://github.com/notional-labs/Tendermint_issues/blob/main/Attack_on_sentinel.md

Output during the attack repeatedly shows the effect of 336 block parts:

```
7:05PM INF Peer ProposalBlockPartSetHeader mismatch, sleeping
blockPartSetHeader={"hash":"15D4BC6DF45AB6A42002DE7F77C492D98BA9FC4D17C1A0F
D8A544AA365D44489","total":1} height=3393430 module=consensus
peer={"Data":{},"Logger":{}}
peerBlockPartSetHeader={"hash":"AE74171AA9816668A59C8B0B5CE1CA337A49FA0420EA
D47A29FA61D04162ACFD","total":336}
7:05PM INF Peer ProposalBlockPartSetHeader mismatch, sleeping
blockPartSetHeader={"hash":"15D4BC6DF45AB6A42002DE7F77C492D98BA9FC4D17C1A0F
D8A544AA365D44489","total":1} height=3393430 module=consensus
peer={"Data":{},"Logger":{}}
peerBlockPartSetHeader={"hash":"AE74171AA9816668A59C8B0B5CE1CA337A49FA0420EA
D47A29FA61D04162ACFD","total":336}
7:05PM INF Peer ProposalBlockPartSetHeader mismatch, sleeping
blockPartSetHeader={"hash":"15D4BC6DF45AB6A42002DE7F77C492D98BA9FC4D17C1A0F
D8A544AA365D44489","total":1} height=3393430 module=consensus
peer={"Data":{},"Logger":{}}
```

```
peerBlockPartSetHeader={"hash":"AE74171AA9816668A59C8B0B5CE1CA337A49FA0420EA
D47A29FA61D04162ACFD","total":336}
7:05PM INF Peer ProposalBlockPartSetHeader mismatch, sleeping
blockPartSetHeader={"hash":"15D4BC6DF45AB6A42002DE7F77C492D98BA9FC4D17C1A0F
D8A544AA365D44489","total":1} height=3393430 module=consensus
peer={"Data":{},"Logger":{}}
peerBlockPartSetHeader={"hash":"AE74171AA9816668A59C8B0B5CE1CA337A49FA0420EA
D47A29FA61D04162ACFD","total":336}
7:05PM INF Peer ProposalBlockPartSetHeader mismatch, sleeping
blockPartSetHeader={"hash":"15D4BC6DF45AB6A42002DE7F77C492D98BA9FC4D17C1A0F
D8A544AA365D44489","total":1} height=3393430 module=consensus
peer={"Data":{},"Logger":{}}
peerBlockPartSetHeader={"hash":"AE74171AA9816668A59C8B0B5CE1CA337A49FA0420EA
D47A29FA61D04162ACFD","total":336}
```

--Mitigation[not a patch/solution]

Recommendations for mitigation:

Required
-[default] maximum block size of 1mb

Optional, recommended for improved mitigation
-Require nonzero gas
-limit IBC memo size to 50kb

In addition to fixes made by Sergio and Ethan Bunchman from Informal, reducing the default maximum block size is the most universal temporary fix. 22mb blocks are much larger than is used by most Cosmos blockchains and should not be set as default even though it is feasible. Cosmos chains can harden from these types of p2p storms by considering the following actions until a patch is available:

-Put a migration in the next chain upgrade that reduces the maximum block size to 5mb, reducing storm impacts by ~75%
-Don’t use sentry node architecture
-Recommend validators QOS traffic to known peers, mildly deprioritizing network traffic from untrusted peers and preventing them from reaching bandwidth peaks above 500mbps

Testnet POCs showing 500-1.5GB throughput of network activity during an unoptimized attack test implies that TOS and other traffic stability tools like VPNs could slow the issue but would only partially mitigate impact as an attack ramps up.

--Proposed Mitigation Alternatives, ie Temporary Solutions

Other theoretical improvements to address this have been considered, understanding that larger configuration in later releases would be ideal to increase security without undue constraints. They may also individually be seen as mitigations, addressing overlap in sdk, ibc and other scopes, but there is not one patch that can address this while keeping existing blockspace as is for the majority of Cosmos SDK blockchains.

Price of bytes must go up

- add governance to x/globalfee so that chains elect their relayers
  - Anyone should be able to make ibc infrastructure transactions in case all the relayers who are elected are hit by a bus
    - These will have much higher prices'
    - Permissionlessness maintained, while making dos very expensive

Limitation on p2p volume for mempool

-
  - Baseline
  - Whitelist with higher limits
- Remove mempool altogether
  - Only gossip proposed blocks
  - “Mempool is broken today and can’t be fixed”
  - Validators should run “intake nodes” that have whitelists they control
- P2P: use qck

Reduce block sizes to cover 90% of the problem temporarily (commonly accepted as safe compared to recorded baselines)

2mb block size and 10 second proposal timeouts (on testnet, but not thoroughly verified against amplification attack testing)

- Lower block size to somewhere between 5-10MB pending numbers
- Increase the gas cost per tx size byte from the vastly underpriced `tx_size_cost_per_byte: "10"` to perhaps 50 to 100
- Consider mempool policy filters for a patch release
  -
    - Apply the size constraint to everything but client updates and contract uploads
      - Problem: can still spin up 1000 single node chains and do a client update ddos machine
- Look for cometBFT problems
- Make Client updates smaller so we can afford to make them
  - Delete:
  - - Signature/block_id_flag, infer "BLOCK_ID_COMMIT"
  - - Signature/Timestamp -> UnixNano timestamp
  - Can't tell if this validator set hash is verified against root, if so nothing that reads as a trivial change then (because priority changes every block), you can make an ID system for validator addrs